SIMULIA

3DEXPERIENCE

ABAQUS 2025 FD02
OUTPUT GUIDE

DASSAULT SYSTEMES

Contents

Trademarks and Legal Notices

Abaqus Output Guide

The Abaqus Output Guide describes how to obtain output from Abaqus and the format of the results (`.fil`) file. It also describes the output variable identifiers that are available.

This guide is a part of the Abaqus® documentation collection, which describes all the capabilities of the Abaqus finite element analysis technology used in SIMULIA® applications.

Abaqus can create the following output files during an analysis:

- a data file containing printed output of the model and history definition generated by the analysis input file processor and, in Abaqus/Standard, printed output of results written during the analysis run;
- an ODB output database file containing results for postprocessing with the Visualization module of Abaqus/CAE (Abaqus/Viewer) and, in Abaqus/Standard, diagnostic information;
- a SIM database file containing results for high-performance postprocessing with the Physics Results Explorerapp on the **3D**EXPERIENCE platform;
- a selected results file in Abaqus/Explicit;
- a results file containing results for postprocessing with external software in Abaqus/Standard and Abaqus/Explicit (in Abaqus/Explicit this file is generated by converting the selected results file);
- a message file containing diagnostic messages about the solution in Abaqus/Standard and Abaqus/Explicit; and
- a status file containing information about the status of the analysis and, in Abaqus/Explicit, diagnostic messages and information about the stable time increment.

What's New

This page describes recent changes in Abaqus Output.

2025 FD01

New Output Variables

New variables are available.

Benefits: The new output variables expand the capabilities of Abaqus/Standard and Abaqus/Explicit.
Total Energy Output Quantities

Output Variable	Field	History	.fil	.dat	Product
ALLERPWR Equivalent radiated power emitted by a panel.	no	yes	no	no	Abaqus/Standard

Nodal Variables

Output Variable	Field	History	.fil	.dat	Product
ERPAC Complex-valued acoustic pressure on the radiating element-based structural surface.	yes	yes	no	no	Abaqus/Standard
ERPWR Equivalent radiated power on an element-based structural surface.	yes	yes	no	no	Abaqus/Standard
ERPWRDEN Equivalent radiated power density on an element-based structural surface.	yes	yes	no	no	Abaqus/Standard

For more information, see *Nodal Variables* and *Whole and Partial Model Variables*.

PDF Guides Available

You can download PDF versions of the Abaqus guides from each guide overview.

Benefits: You can easily access the PDF versions of the guides.
In earlier releases, PDF versions of the guides were updated only for the GA (General Availabilty) release and could be downloaded from the Dassault Systèmes Knowledge Base. As of 2025 FD01, the PDF guides will be updated with each release of the HTML versions of the guides.

For more information, see *Abaqus Output Guide*.

2024 FD03

New Output Variables

New variables are available.

Benefits: The new output variables expand the capabilities of Abaqus/Standard and Abaqus/Explicit.
Element Integration Point Variables

Output Variable	Field	History	.fil	.dat	Product
SOC_*i* State of charge in particle *i* (1 *i* 3).	yes	yes	no	no	Abaqus/Standard

Element Section Variables

Output Variable	Field	History	.fil	.dat	Product
UVARPT Element user-defined output variables.	yes	yes	no	no	Abaqus/Standard
UVARPT*n* Element user-defined output variable *n*.	yes	yes	no	no	Abaqus/Standard

Whole Element Variables

Output Variable	Field	History	.fil	.dat	Product
UVARE Element user-defined output variables.	yes	yes	no	no	Abaqus/Standard
UVARE*n* Element user-defined output variable *n*.	yes	yes	no	no	Abaqus/Standard

Whole and Partial Model Variables

Output Variable	Field	History	.fil	.dat	Product
SOC	no	yes	no	no	Abaqus/Standard

Output Variable	Field	History	.fil	.dat	Product
Current state of charge in the specified region of the model.					

For more information, see *Element Integration Point Variables*, *Whole Element Variables*, and *Whole and Partial Model Variables*.

2024 FD02

New Output Variables

New variables are available.

Benefits: The new output variables expand the capabilities of Abaqus/Standard and Abaqus/Explicit.
Element Integration Point Variables

Output Variable	Field	History	.fil	.dat	Product
PORVAVG Equivalent pore pressure, computed as a volume fraction weighted average of all materials in the element.	yes	no	no		Abaqus/Explicit
SDEFRES Deformation resistance in Anand's creep model.	yes	yes	no	no	Abaqus/Standard
TG Glass transition temperature.	yes	yes	no	no	Abaqus/Standard
TGTDIFF The difference between the glass transition temperature and the temperature.	yes	yes	no	no	Abaqus/Standard

Element Section Variables

Output Variable	Field	History	.fil	.dat	Product
SK	yes	yes	yes	yes	Abaqus/Standard

Output Variable	Field	History	.fil	.dat	Product
Section curvature change and twist components.					

For more information, see *Element Integration Point Variables*, *Element Section Variables*, and *Element Integration Point Variables*.

2024 FD01

New Output Variables

New variables are available.

Benefits: The new output variables expand the capabilities of Abaqus/Standard and Abaqus/Explicit.
Element Integration Point Variables

Output Variable	Field	History	.fil	.dat	Product
MSTRESSCRT	yes	yes	no	no	Abaqus/Standard
Maximum stress-based damage initiation criterion.	no	yes	no		Abaqus/Explicit
MSTRAINCRT	yes	yes	no	no	Abaqus/Standard
Maximum strain-based damage initiation criterion.	no	yes	no		Abaqus/Explicit
TSAIWUCRT	yes	yes	no	no	Abaqus/Standard
Tsai-wu stress-based damage initiation criterion.	no	yes	no		Abaqus/Explicit
TSAIWUECRT	yes	yes	no	no	Abaqus/Standard
Tsai-wu strain-based damage initiation criterion.	no	yes	no		Abaqus/Explicit
DMIFI	yes	yes	no	no	Abaqus/Standard
All active components of the damage initiation failure indices.	yes	yes	no		Abaqus/Explicit
MSTRESSFI	yes	yes	no	no	Abaqus/Standard
Maximum stress-based damage	no	yes	no		Abaqus/Explicit

Output Variable	Field	History	.fil	.dat	Product
initiation failure index.					
MSTRAINFI Maximum strain-based damage initiation failure index.	yes	yes	no	no	Abaqus/Standard
	no	yes	no		Abaqus/Explicit
TSAIWUFI Tsai-Wu stress-based damage initiation failure index.	yes	yes	no	no	Abaqus/Standard
	no	yes	no		Abaqus/Explicit
TSAIWUEFI Tsai-Wu strain-based damage initiation failure index.	yes	yes	no	no	Abaqus/Standard
	no	yes	no		Abaqus/Explicit
TSINVMTCRT Transversely isotropic stress invariant–based matrix tensile damage initiation criterion.	yes	yes	no	yes	Abaqus/Standard
TSINVMCCRT Transversely isotropic stress invariant–based matrix compression damage initiation criterion.	yes	yes	no	yes	Abaqus/Standard
THEFL Thermal strain in the pore fluid in a porous medium.	yes	yes	no	no	Abaqus/Standard
FVE All tensor components of all field expansion strain tensors.	yes	yes	no	no	Abaqus/Standard
FVEFL	yes	yes	no	no	Abaqus/Standard

Output Variable	Field	History	.fil	.dat	Product
All field expansion strains in the pore fluid in a porous medium.					
FVE*n* All tensor components for the field expansion strain tensor due to field variable number *n*.	yes	yes	no	no	Abaqus/Standard
FVEFL*n* Field expansion strain in the pore fluid in a porous medium due to field variable number *n*.	yes	yes	no	no	Abaqus/Standard

Element Face Variables

Output Variable	Field	History	.fil	.dat	Product
CWEAR Accumulated scalar nodal contact wear distance.	yes	yes	no	no	Abaqus/Standard

Surface Variables

Output Variable	Field	History	.fil	.dat	Product
AMBIENTTEMP Reference ambient temperature on element faces from prescribed boundary radiation.	yes	no	no	no	Abaqus/Standard

For more information, see *Element Integration Point Variables* and *Element Integration Point Variables*.

2024 GA

New Output Variables

New variables are available.

Benefits: The new output variables expand the capabilities of Abaqus/Standard and Abaqus/Explicit.

Nodal Variables

Output Variable	Field	History	.fil	.dat	Product
STIFN Local normal stiffness.	yes	no	no	no	Abaqus/Standard
VN Complex-valued surface normal velocity.	yes	yes	no	no	Abaqus/Standard—available only for steady-state (frequency domain) dynamic analyses (modal and direct)
VNSQ Real-valued surface normal velocity squared.	yes	yes	no	no	Abaqus/Standard—available only for steady-state (frequency domain) dynamic analyses (modal and direct)
AVNSQ Area-weighted surface normal velocity squared or acoustic power normalized by the acoustic impedance of the surrounding fluid.	yes	yes	no	no	Abaqus/Standard—available only for steady-state (frequency domain) dynamic analyses (modal and direct)
FEXT All components of external point loads from a co-simulation or external field definition.	yes	yes	no		Abaqus/Explicit
MEXT All components of external point moments from a co-simulation or external field definition.	yes	yes	no		Abaqus/Explicit

Structural Optimization Variables

Output Variable	Field	History	.fil	.dat	Product
DISP_NORMAL_VAL The value of the bead optimization		no	no	no	Abaqus/Standard—output automatically during shape optimization

Output Variable	Field	History	.fil	.dat	Product
displacement along the node normal vector.					

For more information, see *Abaqus/Standard Nodal Variables*, *Structural Optimization Variables*, and *Abaqus/Explicit Nodal Variables*.

About Output

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *Output to the Data and Results Files*
- *Output to the Output Database*
- *Abaqus/Standard Output Variable Identifiers*
- *Abaqus/Explicit Output Variable Identifiers*
- *Diagnostic printing*
- *Degree of freedom monitor requests*

Overview

Abaqus can create the following output files during an analysis:

- a data file containing printed output of the model and history definition generated by the analysis input file processor and, in Abaqus/Standard, printed output of results written during the analysis run;
- an ODB output database file containing results for postprocessing with the Visualization module of Abaqus/CAE (Abaqus/Viewer) and, in Abaqus/Standard, diagnostic information;
- a SIM database file containing results for high-performance postprocessing with the Physics Results Explorerapp on the **3D**EXPERIENCE platform;
- a selected results file in Abaqus/Explicit;
- a results file containing results for postprocessing with external software in Abaqus/Standard and Abaqus/Explicit (in Abaqus/Explicit this file is generated by converting the selected results file);
- a message file containing diagnostic messages about the solution in Abaqus/Standard and Abaqus/Explicit; and
- a status file containing information about the status of the analysis and, in Abaqus/Explicit, diagnostic messages and information about the stable time increment.

Abaqus can create files for restarting an analysis—see *Restarting an Analysis*. In Abaqus/Standard these files can also be used to extract results output not requested during an analysis.

The Data File

The data file (*job-name*`.dat`) is a text file that contains information about the model definition (generated by the analysis input file processor) and, in Abaqus/Standard, tabular output of results. The analysis input file processor information includes the model definition, the history definition, and messages identifying any error and warning conditions that were detected while processing the input data.

Controlling the Amount of analysis input file processor Information Written to the Data File

You can control the amount of information written to the data file by the analysis input file processor in Abaqus/Standard and Abaqus/Explicit.

Input File Usage: Use the following option in the model definition section of the input file:

**PREPRINT*

Abaqus/CAE Usage: Job module: job editor: **General**: **Preprocessor Printout**

Input File Echo

By default, the input file will not be echoed to the data file. You can choose to activate this printout. If the input file is defined in terms of an assembly of part instances, the echo to the data file will be that of the flattened input file (that is, one that does not use parts and assemblies).

Input File Usage: **PREPRINT*, ECHO=YES *or* NO

Abaqus/CAE Usage: Job module: job editor: **General**: **Preprocessor Printout**: **Print an echo of the input data**

Input Parameter Information

For parametrized input files, information about input parameters and their values can be printed in the data file. By default, the modified version of the original input file showing this information will not be printed in the data file. You can choose to activate this printout.

Input File Usage: **PREPRINT*, PARVALUES=YES *or* NO

Abaqus/CAE Usage: Parametrized input files are not supported in Abaqus/CAE.

Parameter-Free Input File Information

For parametrized input files, a parameter-free version (after parameter evaluation and substitution) of the original input file can be printed in the data file. By default, this modified version of the input file will not be printed in the data file. You can choose to activate this printout.

Input File Usage: **PREPRINT*, PARSUBSTITUTION=YES *or* NO

Abaqus/CAE Usage: Parametrized input files are not supported in Abaqus/CAE.

Model and History Definition Summaries

By default, the options defining the model and history data will not be summarized in the data file. You can choose to activate this printout.

For an Abaqus/Explicit analysis the model summary data, when requested, includes the mass, center of mass, and the rotary inertia information for the element sets in the model and for the whole model. However, for two-dimensional models the reported rotary inertia includes the I_{33} component corresponding to the only active rotation degree of freedom; the remaining components are not included.

Input File Usage: **PREPRINT*, MODEL=YES *or* NO, HISTORY=YES *or* NO

Abaqus/CAE Usage: Job module: job editor: **General**: **Preprocessor Printout**: **Print model definition data** and **Print history data**

Contact Constraint Information

In Abaqus/Standard you can choose to activate printout of detailed information about the contact constraints generated by the contact pair definition data.

Input File Usage: **PREPRINT*, CONTACT=YES *or* NO

Abaqus/CAE Usage: Job module: job editor: **General**: **Preprocessor Printout**: **Print contact constraint data**

Mass Information

In Abaqus/Explicit you can choose to activate printout of detailed information about the mass property of each user-defined element set.

Input File Usage: **PREPRINT*, MASS PROPERTY=YES *or* NO

Abaqus/CAE Usage: This parameter is not supported by Abaqus/CAE.

Requesting Printed Results

In Abaqus/Standard the values of output variables can be printed to the data file in tabular format throughout the analysis. You can control the following types of printed output during the analysis run: element output, node output, contact surface output, energy output, fastener interaction output, modal output, section output, and radiation output—see *Output to the Data and Results Files* and *Cavity Radiation in Abaqus/Standard*. You specify the variables to be printed in each output table and, for element variables, the locations at which they are to be printed (at the integration points, at the element centroid, at the nodes, or averaged at the nodes). Nodal variables at nodes with transformations can be written in either the global or the local coordinate system (see *Transformed Coordinate Systems*). The list of available variables is given in *Abaqus/Standard Output Variable Identifiers*. Output of results to the data file is requested as part of a step definition.

Viewing Part and Assembly Information in the Data File

An Abaqus model can be defined in terms of an assembly of part instances (see *Assembly Definition*). In such a model node and element numbers can be repeated within the definitions of different parts. These local numbers are converted internally by Abaqus to unique global numbers, and the output written to the data file is given in terms of those internal numbers. A map between user-defined numbers and internal numbers is printed to the data file (after the step data) if any output that includes node and element numbers is requested in the data file.

Set and surface names that appear in the data file are prefixed by the assembly and part instance names, separated by underscores (`Assembly_Part1-1_setname`, for example).

Local coordinate systems defined within a part or part instance are translated and rotated according to the positioning data given in the part instance definition.

The Output Database

The output database is a neutral binary file. Unlike the restart or binary results files, it can be copied directly from one computing platform to another without translation.

Format of the Output Database

The Abaqus output database is available in two formats, ODB and SIM. By default, the results output is created in ODB format. For an Abaqus/Standard or Abaqus/Explicit analysis you have the option to write results in both formats during the same job. Only results in SIM format can be imported into the **3D**EXPERIENCE platform for high-performance postprocessing. For more information, see *Limitations When Writing and Postprocessing Results in SIM Format* below.

- The ODB output database (*job-name*`.odb`) is used to store model information and analysis results in terms of an assembly of part instances. The Visualization module of Abaqus/CAE (Abaqus/Viewer) uses this output database for postprocessing analysis results and viewing diagnostic information.
- The SIM database file (*job-name*`.sim`) contains model and results information. The Physics Results Explorerapp on the **3D**EXPERIENCE platform uses this database for high-performance postprocessing of analysis results.

Input File Usage: Use the following command line options to write results in an Abaqus/Standard or Abaqus/Explicit analysis in SIM format:

abaqusjob=*job-name* **resultsformat**=`sim`

Use the following command line options to write results in an Abaqus/Standard or Abaqus/Explicit analysis in ODB format and in SIM format:

abaqusjob=*job-name* **resultsformat**=`both`

Abaqus/CAE Usage: Use the following input to write results in an Abaqus analysis in SIM format:

Job module: job editor: **General**: **Results Format**: **SIM**

Use the following input to write results in an Abaqus/Standard or Abaqus/Explicit analysis in ODB format and in SIM format:

Job module: job editor: **General**: **Results Format**: **Both**

Handling of Floating Point Data

By default, floating point data are written in single precision to the ODB output database file. You can choose to write floating point nodal field output data to the ODB output database file in double precision; see *Abaqus/Standard and Abaqus/Explicit Execution* for details.

For Abaqus/Standard and Abaqus/Explicit analyses, floating point data are written to the SIM database in single precision, with the exception of nodal coordinates, which are written in double precision.

Opening an Output Database in Abaqus/CAE

You can open an output database file from an older release of Abaqus in Abaqus/CAE. Output database files from previous releases of Abaqus must be converted to the current release when they are opened. If you are using an older release of Abaqus/CAE, you cannot open an output database file created from a newer release of Abaqus.

Choosing an Output Format

Your choice of output format depends on your level of experience with high-performance visualization, the Physics Results Explorerapp, and your postprocessing needs.

- If you are still learning to use high-performance visualization and you want to compare your results with Abaqus/Viewer, write results in both formats.
- If the model is large and you need the improved performance of the Physics Results Explorerapp, as well as the capabilities of Abaqus/Viewer, write results in both formats.
- If you are confident that the high-performance visualization features in the Physics Results Explorerapp provide all the capabilities you need, write results in SIM format.

Requesting Output to the Output Database

You choose the variables to be written to the output database from the lists in *Abaqus/Standard Output Variable Identifiers* and *Abaqus/Explicit Output Variable Identifiers*. The following types of output are available: element output, node output, contact surface output, energy output, integrated output, time incrementation output, fastener interaction output, modal output, and radiation output. In addition, a subset of the diagnostic information that is written to the message file in Abaqus/Standard and Abaqus/Explicit (see *The Message File in Abaqus/Standard and Abaqus/Explicit*) and to the Abaqus/Explicit status file (see *The Status File*) is included in the output database. See *Output to the Output Database* for a detailed explanation of how to generate output database requests.

Three types of information are stored in the output database: “field” output, “history” output, and diagnostic information. Field output is intended to be relatively infrequent output for a large portion of the model.Abaqus/CAE uses field output to generate contour plots, displaced shape plots, symbol plots, and *X–Y* plots in the Visualization module. History output is intended to be output for a small portion of the model requested at a fairly high frequency.Abaqus/CAE uses history output to generate *X–Y* plots in the Visualization module.See *Output to the Output Database* for detailed descriptions of field and history output. Diagnostic information is intended to provide convergence information for use in Abaqus/CAE; for more information, see *Viewing diagnostic output*.

Limitations When Writing and Postprocessing Results in SIM Format

A subset of options in Abaqus/Standard and Abaqus/Explicit are not supported for analyses that produce results in SIM format. If you include one or more of these options or parameters in your analysis and write output in SIM format or both formats, the analysis will either terminate with errors or produce limited results.

The following options produce error messages in the data (`.dat`) file:

```
*ADAPTIVE MESH REFINEMENT
*CONTOUR INTEGRAL
*DIRECT CYCLIC, FATIGUE
*ELECTROMAGNETIC
*ENRICHMENT
*ENRICHMENT ACTIVATION (for XFEM)
*IMPORT
```

```
*MAP SOLUTION
*MAGNETOSTATIC
*NMAP, FATIGUE=BLENDED or TOROIDAL
*POST OUTPUT
*REBAR
*SUBSTRUCTURE PATH
*SURFACE, TYPE=(EULERIAN MATERIAL, XFEM, BSPLINE, BEZIER, or USER)
*STEADY STATE TRANSPORT
*SYMMETRIC MODEL GENERATION
*SYMMETRIC RESULTS TRANSFER
*TRACER PARTICLE
```

The following option produces limited results but no error messages:

```
*EULERIAN SECTION: some volume fraction data are not written to the SIM database
```

In addition, the following option produces results in SIM format; however, the results are not accounted for in the Physics Results Explorerapp:

```
*MODEL CHANGE
```

The Selected Results File

The Abaqus/Explicit selected results file (*job-name*`.sel`) stores user-selected results, which are converted into the results file (*job-name*`.fil`) for postprocessing by other commercial postprocessing packages.

Element output, node output, and energy output can be requested (see *Output to the Data and Results Files* for details); the variables available for output are listed in *Abaqus/Explicit Output Variable Identifiers*. You can write a user-selected subset of the results for a given node set or element set at more frequent intervals than the restart intervals. You specify the output requests within a step definition, which allows you to be selective about the amount of data written to the selected results file to avoid using excessive disk storage. For example, when dealing with a very large model, you may choose to write only the current displacements and the equivalent plastic strain for the entire model 20 times in the step and to write the acceleration history at one node 200 times in the step.

The Results File

The Abaqus results file in Abaqus/Standard and Abaqus/Explicit (*job-name*`.fil`) can be read by external postprocessors to produce *X–Y* plots or printed tabular output. Most commercial finite element results-display packages provide translators that use the Abaqus results file as their input. The results file can also be used as a convenient medium for importing analysis results into your own postprocessing program. *Accessing the Results File Information* provides details on how to read this file.

Results file output of temperature from a heat transfer, thermal-electrical, or thermal-electrical-structural analysis can be used as input to a stress analysis of the same mesh (see *Sequentially Coupled Thermal-Stress Analysis*).

Obtaining Results File Output in Abaqus/Standard

In Abaqus/Standard you choose the variables to be written to the results file from the lists in *Abaqus/Standard Output Variable Identifiers* in a manner similar to that for output printed to the data file. You must specifically request that values be written to the results file or none will be provided. Element output, node output, contact surface output, energy output, modal output, and radiation output are available—see *Output to the Data and Results Files* and *Cavity Radiation in Abaqus/Standard* for details.

Obtaining Results at the Beginning of a Step

You can request that the solution state at the beginning of a step (the zero increment) be written to the Abaqus/Standard results file. Zero-increment file output is available only for steps in which the concept of time governs the incrementation scheme of the selected procedure and, hence, the following procedures are excluded:

- Linear static perturbation analysis (*Static Stress Analysis*)
- *Eigenvalue Buckling Prediction*
- *Natural Frequency Extraction*
- *Mode-Based Steady-State Dynamic Analysis*
- *Response Spectrum Analysis*
- *Random Response Analysis*

If you request zero-increment results file output, it will be generated for all valid procedures in a given analysis.

You must request zero-increment results file output to generate a zero-increment results file in a data check analysis (see *Abaqus/Standard and Abaqus/Explicit Execution*). It is strongly recommended that you request zero-increment results file output if the results file is used to drive a submodel; see *Node-Based Submodeling* for further discussion.

Input File Usage: **FILE FORMAT*, ZERO INCREMENT

The **FILE FORMAT* option can be given as model data or as history data, but it can appear only once in the input file.

Abaqus/CAE Usage: Results file output cannot be requested in Abaqus/CAE.

Obtaining Results File Output in Abaqus/Explicit

The Abaqus/Explicit results file is a sequential access file generated from the selected results file (see *Abaqus/Standard and Abaqus/Explicit Execution*). The results file contains the requested results in the format described in *Results File*.

Input File Usage: Use either of the following command line options to convert a selected results file to a results file:

abaqusjob=*job-name* **convert**=`select`

abaqusjob=*job-name* **convert**=`all`

Abaqus/CAE Usage: The selected results file cannot be converted in Abaqus/CAE.

Part and Assembly Information

An Abaqus model can be defined in terms of an assembly of part instances (see *Assembly Definition*). However, the results file does not contain part and assembly records.

In a model defined in terms of an assembly of part instances, node and element numbers can be repeated within the definitions of different parts. These local numbers are converted internally by Abaqus to unique global numbers, and the output written to the results file is given in terms of the global (internal) numbers. A map between user-defined numbers and internal numbers is printed to the data file if any results file output that includes node and element numbers is requested.

Set and surface names that appear in the results file are prefixed by the assembly and part instance names, separated by underscores (`Assembly_Part1-1_setname`, for example).

Local coordinate systems defined within a part or part instance are translated and rotated according to the positioning data given in the part instance definition.

Format of the Results File

The Abaqus results file in Abaqus/Standard or Abaqus/Explicit is organized as a sequential file, in binary or in ASCII format. ASCII format is necessary if the file is to be read on a computer system that is different from the one on which the file was written. ASCII format allows the results file to be transferred between different computer systems without having to translate binary data. ASCII format is not needed if the file will always be used on the same system or on systems that use the same binary format. If the results file output will always reside on the same computer, the default binary format is usually the most efficient way of storing the file. For large problems a file in ASCII format will be significantly larger than the same file in binary format.

Controlling the Format of the Results File in Abaqus/Standard

Abaqus/Standard can write the results file in either binary or ASCII format. The default format is binary.

The results file output must be written in the same format for the entire analysis. The format cannot be changed upon restarting the problem.

The format of the Abaqus/Standard results file can also be controlled in the Abaqus/Standard environment file (see *Environment File Settings*). The format specified in an analysis supersedes the value defined in the environment file.

In addition, the **ascfil** facility in the Abaqus execution procedure (*ASCII Translation of Results (.fil) Files*) can be used to convert a binary Abaqus/Standard results file (*job-name*`.fil`) to ASCII format (*job-name*`.fin`) after the analysis completes.

Input File Usage: **FILE FORMAT*, ASCII

The **FILE FORMAT* option can be given as model data or as history data, but it can appear only once in the input file.

Abaqus/CAE Usage: Results file output cannot be requested in Abaqus/CAE.

Controlling the Format of the Results File in Abaqus/Explicit

Abaqus/Explicit always writes the results file output in binary format during file conversion, but the binary Abaqus/Explicit results file can be converted to ASCII format using the **ascfil** facility (*ASCII Translation of Results (.fil) Files*).

ASCII Format

Results File defines the contents of the records that are written to the results file; these descriptions also hold if the results file is written in ASCII format. All the data items in these files are either integers, floating point numbers, or character strings. When ASCII format is requested, each data item is translated into an equivalent character string before it is written to the file. These strings are written in 80-character logical records in the order described in the record definitions.

Each 80-character logical record is completely filled before the next one is started, so that any data item can be split, with some of the characters that define the item in one logical record and the remainder in the next. Each data item usually follows immediately behind its predecessor. The exception is that for results file record key 2001 Abaqus will fill out the logical record with blank characters, so that the record can be written immediately to the physical storage medium. Abaqus then inserts a logical record consisting of 80 blanks, which allows the end-of-file to be handled correctly.

The beginning of each "record" is indicated by an asterisk (*). Each floating point number begins with the character D, followed by the number in the format E22.15 or D22.15, depending on whether the release of Abaqus that wrote the results file used single precision or double precision. Each character string begins with the character A, followed by eight characters (if the character string has fewer than eight characters, the right part of the string is blank; character strings longer than eight characters are written eight characters at a time). Each integer begins with the character I, followed by a two digit integer giving the number of decimal digits in the integer, followed by the integer itself (written as decimal digits).

For example, record key 1900 for an S4R element with element number 5 and nodes 195, 198, 205, and 204 would be written

```
*I 18I 41900I 15AS4R     I 3195I 3198I 3205I 3204
```

and record key 101 for node 135 and 6 degrees of freedom would be written

```
*I 19I 3101I 3135D1.280271914214298E-10D1.500000000000036E+00
D-1.074629835784448E-46D 6.983222716550941E-12
D-4.084928798492785E-13D-1.072688441364597E-10
```

Precision of Floating Point Data in the Results File

The precision of floating point data written to the results file depends on the precision of the executable that generates the data. Abaqus/Standard always uses double precision; thus, floating point data are always written to the Abaqus/Standard results file in double precision. Abaqus/Explicit can be run in single or double precision on most machines; see *Defining an Analysis* for details on the precision level of the Abaqus/Explicit executable. If the double precision executable for Abaqus/Explicit is used, floating point data are written to the Abaqus/Explicit results file in double precision; likewise, if the single precision executable for Abaqus/Explicit is used, floating point data are written to the Abaqus/Explicit results file in single precision.

Maximizing the Efficiency of the Results File

In Abaqus/Standard each element output request (a collection of identifying keys entered on a single line) is preceded by an "element header" record (see *Results File*). Hence, the size of the results file can be minimized by entering all element output variables of the same "type" (element integration point variable, element section variable, whole element variable, etc.) on a single line. (See *Output to the Data and Results Files* for an explanation of the output variable types.) Consolidating output variable entries is encouraged, since it will reduce the size of the results file.

Example

For example, the following output requests can be used to request output of element variables in the results file in a stress/displacement analysis:

```
*EL FILE
 S, SINV, E, PE, CE, EE, ENER, TEMP, FV, COORD
 SF, SE
 LOADS, ELEN, EVOL
*EL FILE, REBAR
 S, SINV, E, PE, CE, EE, RBFOR, RBANG
 SF, SE
 LOADS, ELEN
```

(The output requests for rebar quantities need not be the same as the underlying element output requests.)

The Message File in Abaqus/Standard and Abaqus/Explicit

The message file (*job-name*`.msg`) is a text file that contains diagnostic messages about the progress of the solution.

The Abaqus/Standard Message File

In Abaqus/Standard the message file contains diagnostic or informative messages about the progress of the solution. If any of these messages describe errors or warnings, the number of such errors or warnings is also given at the end of the data file. The message file is written automatically during an Abaqus/Standard analysis.

The Abaqus/Standard message file contains information about the increment number, step time, fraction of a step completed, equilibrium iterations, severe discontinuity (contact) iterations, plasticity algorithms, adaptive mesh smoothing, the load proportionality factor in a Riks analysis, etc. A portion of the diagnostic information in the message file is also written to the output database for use in Abaqus/CAE (for more information, see *Requesting Output to the Output Database*).

You can control the amount of information written to the message file for each step. This feature is sometimes helpful in difficult analyses since it allows detailed diagnostic information to be written about certain events (such as contact) during a nonlinear solution; this information can often be useful in developing a strategy for the solution of highly nonlinear problems.

Input File Usage: **PRINT*

The **PRINT* option can appear only once within a step definition.

Abaqus/CAE Usage: Step module: **Output->Diagnostic Print**

Controlling the Frequency of Output to the Message File

You can control the frequency at which information is printed to the message file by specifying the desired output frequency in increments. The default output frequency is 1 (or 10 in a direct cyclic or a low-cycle fatigue analysis). The output will always be printed at the last increment of each step unless you specify a frequency of zero to suppress the output.

Input File Usage: **PRINT*, FREQUENCY=*N*

Abaqus/CAE Usage: Step module: **Output->Diagnostic Print**: **Frequency** *N*

Requesting Detailed Contact Printout

You can obtain a detailed printout of contact conditions during iteration. This information about which points are contacting or separating in interface and gap problems is useful in tracking the development of the solution in difficult contact problems. The details are written for every severe discontinuity iteration. By default, the detailed contact output is suppressed.

Input File Usage: **PRINT*, CONTACT=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Contact**

Requesting Detailed Model Change Printout

You can obtain a detailed printout of model change operations (removal and reactivation) at the start of a step. This information includes the new original coordinates and normals of elements being reactivated strain free in a large-displacement analysis. By default, the detailed model change output is suppressed. See *Element and Contact Pair Removal and Reactivation* for details on model change operations.

Input File Usage: **PRINT*, MODEL CHANGE=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Model Change**

Requesting Detailed Printout of Problems with the Plasticity Algorithms

You can activate printout of element and integration point numbers for which the plasticity algorithms have failed to converge during an iteration. This information is useful for finding the place in the mesh and/or the plasticity model at which Abaqus is encountering material model difficulties. Modeling problems and material parameter specification problems can be identified using this detailed printout. By default, this printout is suppressed.

Input File Usage: **PRINT*, PLASTICITY=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Plasticity**

Requesting Output of Equilibrium Residuals

By default, equilibrium residuals during equilibrium iterations are output. You can choose to suppress this output entirely, but it is not recommended; without the output of equilibrium residuals, you cannot see the accuracy of the iteration process.

Input File Usage: **PRINT*, RESIDUAL=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Residual**

Requesting Solver Information

By default, information about the number of equations being solved and the number of floating point operations is output for each iteration. You can request for this output to be suppressed.

Input File Usage: **PRINT*, SOLVE=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Solve**

Requesting Detailed Adaptive Mesh Smoothing Printout

You can activate detailed printout of adaptive mesh smoothing in Abaqus/Standard. The output includes information about the magnitude of the maximum displacement and the node and degree of freedom where the maximum displacement increment occurs during each mesh sweep. It also provides the node numbers at which geometric feature changes occur. By default, only a summary is output.

Input File Usage: **PRINT*, ADAPTIVE MESH=YES *or* NO

Abaqus/CAE Usage: Adaptive mesh output to the message file is not supported in Abaqus/CAE.

Monitoring a Degree of Freedom in the Message File

You can write the current value of a specified point and degree of freedom to the message file. This information can be used to monitor the progress of the solution. The information will also be written in the status file (see below). You can control the frequency at which the value is printed in the message file. The default frequency is 1 (or 10 in a direct cyclic analysis).

Degree of freedom monitoring does not apply to eigenvalue buckling prediction, eigenfrequency extraction, or response spectrum procedures. For other linear perturbation procedures output for the monitored degree of freedom is the base state value.

Input File Usage: **MONITOR*, NODE=*node_number*, DOF=*dof*, FREQUENCY=*N*

The node and degree of freedom being monitored can be changed from step to step by repeating the **MONITOR* option. The node and degree of freedom specified in the last occurrence of this option in a step will be used for that step.

Abaqus/CAE Usage: Step module: **Output->DOF Monitor**: **Monitor a degree of freedom throughout the analysis**, click **Edit** to select the point, **Degree of freedom:** *dof*, **Print to the message file every *N* increments**

In Abaqus/CAE only one point and degree of freedom can be monitored for an analysis; you cannot change the monitor request from step to step.

The Abaqus/Explicit Message File

In Abaqus/Explicit the message file contains messages if potential problems are detected during an analysis. You can control the output of diagnostic messages for each step (see *Explicit Dynamic Analysis* and *Contact Diagnostics in an Abaqus/Explicit Analysis*). A portion of the diagnostic information in the message file is also written to the output database for use in Abaqus/CAE (for more information, see *Requesting Output to the Output Database*).

The Status File

The status file (*job-name*`.sta`) is a text file that contains information about the progress of an analysis.

The Abaqus/Standard Status File

The Abaqus/Standard status file contains a single 80-character record for each increment and is updated upon completion of each increment of an analysis. This record is written directly to secondary storage immediately at the completion of the increment. Therefore, the status file can be examined as the analysis job is executing, thus providing a monitor of the progress of the analysis. Other than specifying that a degree-of-freedom variable be monitored in the status file in Abaqus/Standard (as described below), the information written to the Abaqus/Standard status file cannot be controlled.

The Abaqus/Explicit Status File

In Abaqus/Explicit the status file (*job-name*.sta) contains, by default, mass and inertial properties for the model, initial stable time increment information, a synopsis of the progress of the analysis including total accumulated CPU usage and the current time increment size, and an estimate of the memory required to process each step. You can control additional output including the total kinetic energy, the energy balance, the identifier of the element with the smallest stable time increment, and the percent change in total mass of the model due to mass scaling.

The frequency at which summary increments are written to the Abaqus/Explicit status file depends on the duration of the analysis in CPU minutes and the amount of output specified in the analysis. The following list provides general guidelines for when a summary increment will be written to the status file.

Summary information will generally be written:

- Each time restart information, field output to the output database, or results file output is written.
- Once per increment if the problem requires fewer than 20 increments.
- 20 times during the step for a short analysis (less than 40 CPU minutes).
- Every 2 CPU minutes for an analysis longer than 40 CPU minutes.

A degree-of-freedom variable can be monitored in the status file while the analysis is running. You can also write additional diagnostic information to the status file (see *Explicit Dynamic Analysis* and *Contact Diagnostics in an Abaqus/Explicit Analysis* for details). A portion of the diagnostic information in the status file, including information for each summary increment, is also written to the output database for use in Abaqus/CAE (for more information, see *Requesting Output to the Output Database*).

Errors that can be detected only while packaging the data for Abaqus/Explicit or during analysis are also written to the status file.

Input File Usage: **PRINT*

The **PRINT* option can appear only once within a step definition.

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**

Requesting Kinetic Energy Output

By default, the kinetic energy for the model is written to the status file. This output is written periodically throughout the step. You can choose to include or exclude the kinetic energy output for each step.

Input File Usage: **PRINT*, ALLKE=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Allke**

Requesting Total Energy Output

By default, the energy balance is written periodically throughout the step. You can choose to include or exclude the energy balance output for each step.

Input File Usage: **PRINT*, ETOTAL=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Etotal**

Requesting Output of the Critical Element

By default, the number of the element with the current minimum stable time increment is output to the status file. This output is written periodically throughout the step. You can choose to include or exclude the critical element output for each step.

Input File Usage: **PRINT*, CRITICAL ELEMENT=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Crit. Elem.**

Requesting Output of the Change in the Total Mass

You can write the percent change in total mass of the model due to mass scaling to the status file for each step. This output is written periodically throughout the step. The percent change in total mass is printed by default only if mass scaling is present in the model.

Input File Usage: **PRINT*, DMASS=YES *or* NO

Abaqus/CAE Usage: Step module: **Output**->**Diagnostic Print**: toggle on **Dmass**

Monitoring a Degree of Freedom in the Status File

You can write the current value of a specified point and degree of freedom to the Abaqus/Standard status file. The value of the point and degree of freedom being monitored will appear in the status file for every increment written during the analysis.

When a degree of freedom is monitored in the Abaqus/Standard status file, the same information is written to the message file (see above), but the specified frequency has no effect on the output to the status file.

Degree of freedom monitoring does not apply to eigenvalue buckling prediction, eigenfrequency extraction, or response spectrum procedures. For other linear perturbation procedures output for the monitored degree of freedom is the base state value.

Input File Usage: **MONITOR*, NODE=*node_number*, DOF=*dof*

The node and degree of freedom being monitored can be changed from step to step by repeating the **MONITOR* option. The node and degree of freedom specified in the last occurrence of this option in a step will be used for that step.

Abaqus/CAE Usage: Step module: **Output**->**DOF Monitor**: **Monitor a degree of freedom throughout the analysis**, click **Edit** to select the point, **Degree of freedom:** *dof*

In Abaqus/CAE only one point and degree of freedom can be monitored for an analysis; you cannot change the monitor request from step to step.

Requesting Output in Multiple Steps

In general, output requests apply to the step in which they are given and to all subsequent steps until they are respecified. However, output specifications for linear perturbation steps (available only in Abaqus/Standard; see below and *General and Perturbation Procedures*) are treated independently of output requests for general analysis steps and apply only to a continuous sequence of linear perturbation steps.

Database output, printed output, and results file output are independent output modes in Abaqus; therefore, changing the specification for one form of output does not affect the other forms.

General Analysis Steps

The default output requests are used in the first general analysis step of an analysis unless you redefine them. For subsequent general analysis steps, the definition of each form of output from the previous general step is maintained unless you redefine it.

Linear Perturbation Steps

The default output requests are used in the first of any sequence of linear perturbation steps unless they are redefined in that step. If a subsequent linear perturbation step is defined without an intermediate general analysis step, the definition of each mode of output from the previous perturbation step is maintained unless you redefine it. If an intermediate general step is defined, the default output requests are again used in the linear perturbation step unless they are redefined in that step.

Element Matrix Output in Abaqus/Standard

In Abaqus/Standard you can write element stiffness matrices and, if available, mass matrices for each step to a file. For heat transfer elements the operator matrices are written if stiffness matrix output is requested.

Element matrix output is available only for elements without internal nodes (unless those nodes have no active degrees of freedom) and with no acoustic or internal degrees of freedom. Examples of elements for which element matrix output is prohibited include acoustic, pipe, elbow, frame, gap, and interface elements as well as axisymmetric elements with Fourier modes. Element matrix output is not available for elements with coupled fields such as coupled temperature-displacement elements and pore pressure elements. For incompatible mode and hybrid elements, stiffness matrix output is prohibited while mass matrix output is available. A substructure matrix output request is used to write a substructure's reduced stiffness matrix, mass matrix, and load case vectors to a file (see *Generating Substructures*).

Element matrix output cannot be requested in a mode-based dynamic analysis (response spectrum, steady-state dynamic, modal dynamic, or random response). However, it can be requested in the eigenfrequency extraction analysis that precedes the mode-based dynamic analysis to output the mass and stiffness matrices.

The element matrices are written without the effects of nodal conditions; therefore, boundary conditions, concentrated loads, and the effects of multi-point constraints are not included in this output. The degrees of freedom are always in the global directions, even if a local coordinate system (*Transformed Coordinate Systems*) has been defined at nodes associated with the element.

You must select the element set for which output is requested. For models defined in terms of an assembly of part instances (*Assembly Definition*), element numbers written with element matrix output are internal numbers generated by Abaqus/Standard. A map between internal numbers and the original element numbers and part instance names is provided in the data file.

Writing the Element Matrices to the Results File

By default, element matrix output records are written to the Abaqus/Standard results file. The record formats for the results file are described in *Results File*. The file can be written in binary or ASCII format based on the file format you specify (see *Controlling the Format of the Results File in Abaqus/Standard* above).

Input File Usage: **ELEMENT MATRIX OUTPUT*, ELSET=*element_set*, OUTPUT FILE=RESULTS FILE

Abaqus/CAE Usage: Element matrix output is not supported in Abaqus/CAE.

Writing the Element Matrices to a User-Defined File

You can write the element matrices to a user-defined file. The file name should not include an extension; the extension `.mtx` will be added. (See *Input Syntax Rules* for the syntax of user-specified file names.)

The format of the output file is compatible with the linear user element (see *User-Defined Elements*).

Input File Usage: **ELEMENT MATRIX OUTPUT*, ELSET=*elset*, OUTPUT FILE=USER DEFINED, FILE NAME=*output_file_name*

Abaqus/CAE Usage: Element matrix output is not supported in Abaqus/CAE.

Writing the Element Matrices to the Data File

You can write the element matrix records to the Abaqus/Standard data file.

Input File Usage: **ELEMENT MATRIX OUTPUT*, ELSET=*elset*, OUTPUT FILE=USER DEFINED

Abaqus/CAE Usage: Element matrix output is not supported in Abaqus/CAE.

Including Distributed Loads

You can choose to write the load vector from distributed loads on the elements. By default, the load vector is not written.

Input File Usage: **ELEMENT MATRIX OUTPUT*, ELSET=*elset*, DLOAD=YES *or* NO

Abaqus/CAE Usage: Element matrix output is not supported in Abaqus/CAE.

Controlling the Frequency of Element Matrix Output

You can control the frequency at which element matrix output will be written by specifying an output frequency in increments. By default, the element matrices will be output every increment (equivalent to an output frequency of 1). Specify an output frequency of 0 to suppress output of the element matrices. Unless the output is suppressed, the matrices will always be written at the last increment of a step.

Input File Usage: **ELEMENT MATRIX OUTPUT*, ELSET=*elset*, FREQUENCY=*N*

Abaqus/CAE Usage: Element matrix output is not supported in Abaqus/CAE.

Writing the Stiffness or Operator Matrix

You can choose to output the stiffness matrix (or operator matrix in heat transfer elements). By default, the stiffness (operator) matrix is not output.

Input File Usage: **ELEMENT MATRIX OUTPUT*, ELSET=*elset*, STIFFNESS=YES *or* NO

Abaqus/CAE Usage: Element matrix output is not supported in Abaqus/CAE.

Writing the Mass Matrix

You can choose to output the mass matrix. By default, element mass matrices are not output.

Input File Usage: **ELEMENT MATRIX OUTPUT*, ELSET=*elset*, MASS=YES *or* NO

Abaqus/CAE Usage: Element matrix output is not supported in Abaqus/CAE.

User-Defined Output Variables in Abaqus/Standard

In Abaqus/Standard output quantities can be defined as functions of any element integration point variable listed in *Abaqus/Standard Output Variable Identifiers* by using user subroutine *UVARM*. Then, output variable UVARM*n* can be requested for output to the data file, the results file, or the output database.

User-Defined State Variables in Abaqus/Standard

In Abaqus/Standard you can allocate solution-dependent state variables and define them in user subroutines defining material behavior, as well as user subroutines *FRIC*, *UEL*, and *UINTER* (see *About User Subroutines and Utilities*). Output variable SDV*n* can be requested for output of these state variables to the data file, the results file, or the output database. For user-defined elements output variable SDV*n* cannot be requested for output to the output database.

Postprocessing with Abaqus/CAE

Abaqus/CAE provides interactive graphical postprocessing from the Abaqus output database file in the Visualization module (also licensed separately as Abaqus/Viewer). Capabilities include model and deformed shape plotting, contour plotting, vector plotting, *X*–*Y* plotting, and animation.

Recovering Additional Results Output from Restart Data in Abaqus/Standard

Data needed for restart in Abaqus/Standard are contained in several files that are generated when you request that restart data be written for an analysis: the restart (`.res`), analysis database (`.mdl` and `.stt`), part (`.prt`), and output database (`.odb`) files. *Restarting an Analysis* describes the writing of restart data in more detail.

In Abaqus/Standard you can extract output from the restart data and write it to new data (`.dat`), results (`.fil`), and output database (`.odb`) files using a postprocessing analysis procedure. If the original analysis included user subroutines, the postprocessing analysis procedure requires the specification of the user subroutines. The data, results, and output database file output requests are defined as described in *Output to the Data and Results Files* and *Output to the Output Database*. The output requests should be defined exactly as they would be in an analysis, except that:

1. The output frequency specification has no meaning and is, therefore, ignored (unless you are recovering additional output from a previous direct cyclic or low-cycle fatigue analysis). Instead, you specify each increment at which output is to be generated in the postprocessing procedure definition.
2. No default output is provided to the output database. Furthermore, model information, such as boundary conditions, is not written to the output database.
3. Element set energy information cannot be recovered since it is not written to the restart file.
4. Output is not available for procedures that do not support restart; for example, linear perturbation procedures.

The element sets and node sets that are defined for the analysis can be used for defining output sets during the postprocessing procedure. Additional sets can also be defined for the postprocessing procedure. You specify the step number in the restart file from which output is required. You cannot obtain results at the beginning of a step (see below).

Input File Usage: **POST OUTPUT*, STEP=*step_number*

When the **POST OUTPUT* option is used, it must appear as the first option in the input file. No data lines from the analysis input file are required. This option can be repeated as often as necessary to obtain further output. Since **POST OUTPUT* is a purely postprocessing procedure, analysis options must not appear in the input file.

Abaqus/CAE Usage: Postprocessing of restart data is not supported in Abaqus/CAE.

Recovering Additional Output from a Direct Cyclic Analysis

If you use this postprocessing technique to recover additional output from a previous direct cyclic analysis (see *Direct Cyclic Analysis*), you must specify the iteration number in the restart file from which output is required instead of the increment. If temperatures (or predefined field variables) are read from a results (`.fil`) file in the original direct cyclic analysis, the same temperatures (or predefined field variables) must be read into the postprocessing analysis. This specification is needed to recover thermal strains at each time increment in the original direct cyclic analysis since the results file is not stored in the restart analysis database.

Input File Usage: **POST OUTPUT*, STEP=*step_number*, ITERATION=*iteration_number*

There are no data lines associated with this option if the ITERATION parameter is specified.

Abaqus/CAE Usage: Postprocessing of restart data is not supported in Abaqus/CAE.

Recovering Additional Output from a Low-Cycle Fatigue Analysis

If you use this postprocessing technique to recover additional output from a previous low-cycle fatigue analysis (see *Low-Cycle Fatigue Analysis Using the Direct Cyclic Approach*), you must specify the cycle number in the restart file from which output is required instead of the increment. If temperatures (or predefined field variables) are read from a results (`.fil`) file in the original low-cycle fatigue analysis, the same temperatures (or predefined field variables) must be read into the postprocessing analysis. This specification is needed to recover thermal strains at each time increment in the original low-cycle fatigue analysis since the results file is not stored in the restart analysis database.

Input File Usage: **POST OUTPUT*, STEP=*step_number*, CYCLE=*cycle_number*

There are no data lines associated with this option if the CYCLE parameter is specified.

Abaqus/CAE Usage: Postprocessing of restart data is not supported in Abaqus/CAE.

Example

A job can be submitted using the following input file. The analysis for which restart data were written must be specified when you submit the job (using the **oldjob** parameter of the Abaqus execution procedure). This example creates a new data (`.dat`) file containing tabular data. The first two tables will contain data from increments 5 and 10 of Step 1 and will give the reaction forces of the nodes in the set `CLAMP`, which was defined when the analysis was run. The next table will contain data from increment 3 of Step 2 and will give displacements from the new node set `TIP` that is defined in this postprocessing analysis.

```
*HEADING
*POST OUTPUT, STEP=1
 5, 10
*NODE PRINT, NSET=CLAMP
 RF,
*POST OUTPUT, STEP=2
 3,
*NSET, NSET=TIP
 1200, 1203, 1205
*NODE PRINT, NSET=TIP
 U,
```

The following example input file recovers additional output from a previous direct cyclic analysis and creates a new output database (`.odb`) file, which contains the stress and strain for the elements in the set `ELIST` from each increment in Iteration 5 of Step 1, followed by data from each increment in Iteration 10 of Step 1:

```
*HEADING
*POST OUTPUT, STEP=1, ITERATION=5
*OUTPUT, HISTORY
*ELEMENT OUTPUT, ELSET=ELIST
 S,E
*POST OUTPUT, STEP=1, ITERATION=10
*OUTPUT, HISTORY
*ELEMENT OUTPUT, ELSET=ELIST
 S,E
```

The following example input file recovers additional output from a previous low-cycle fatigue analysis and creates a new output database (`.odb`) file, which contains the stress and strain for the elements in the set `ELIST` from each increment in Cycle 5 of Step 1, followed by data from each increment in Cycle 10 of Step 1:

```
*HEADING
*POST OUTPUT, STEP=1, CYCLE=5
*OUTPUT, HISTORY
*ELEMENT OUTPUT, ELSET=ELIST
 S,E
*POST OUTPUT, STEP=1, CYCLE=10
*OUTPUT, HISTORY
*ELEMENT OUTPUT, ELSET=ELIST
 S,E
```

Output to the Data and Results Files

Products: Abaqus/Standard Abaqus/Explicit

References:

- *About Output*
- **CONTACT FILE*
- **CONTACT PRINT*
- **EL FILE*
- **EL PRINT*
- **ENERGY FILE*
- **ENERGY PRINT*
- **FILE OUTPUT*
- **MODAL FILE*
- **MODAL PRINT*
- **NODE FILE*
- **NODE PRINT*
- **RADIATION FILE*
- **RADIATION PRINT*
- **SECTION PRINT*
- **SECTION FILE*

Overview

Output variables are available for:

- element integration points, element section points, whole elements, and element sets;
- nodes;
- the whole model;
- modes in mode-based dynamics procedures;
- surfaces in Abaqus/Standard; and
- sections in Abaqus/Standard.

All of the output variables are defined in *Abaqus/Standard Output Variable Identifiers* and *Abaqus/Explicit Output Variable Identifiers*. Output quantities from the elements, nodes, and whole model can be written to the data and results files in Abaqus/Standard and to the selected results file in Abaqus/Explicit. In Abaqus/Standard output quantities from eigenmodes, surfaces, and sections can also be written to the data and results files.

For Abaqus models defined in terms of an assembly of part instances (see *Assembly Definition*), output in the data and results files is given in terms of node, element, set, and surface labels generated internally by Abaqus. See *About Output* for details on how to relate the internally generated numbers and names to those you specified.

Requesting Output to the Data and Results Files

The following sections discuss the input file syntax for requesting output to the data and results files. Abaqus/CAE automatically requests that a data file containing the default printed output for the current analysis procedure at the end of each step be generated; you cannot control the contents of the data file from within Abaqus/CAE. An analysis from Abaqus/CAE does not create a results file.

Output to the Abaqus/Standard Data File

Abaqus/Standard analysis results can be written to the data (`.dat`) file. Element output, nodal output, contact surface output, energy output, modal output, and section output are available.

Input File Usage: Use any of the following options to request output to the Abaqus/Standard data file:

**CONTACT PRINT*
**EL PRINT*
**ENERGY PRINT*
**MODAL PRINT*
**NODE PRINT*
**SECTION PRINT*

These options are discussed in detail below.

Output to the Abaqus/Standard Results File

Abaqus/Standard analysis results can be written to the results (`.fil`) file. Element output, nodal output, contact surface output, energy output, modal output, and section output are available.

Input File Usage: Use any of the following options to request output to the Abaqus/Standard results file:

**CONTACT FILE*
**EL FILE*
**ENERGY FILE*
**MODAL FILE*
**NODE FILE*
**SECTION FILE*

These options are discussed in detail below.

Output to the Abaqus/Explicit Results File

You can write Abaqus/Explicit analysis results to the selected results (`.sel`) file by specifying a results file output request in conjunction with element output, nodal output, and/or energy output requests, as explained below. A results file output request can appear only once per step but remains in effect in subsequent steps unless it is redefined.

You can convert the selected results file (*job-name*`.sel`) into the results (*job-name*`.fil`) file using the **convert** utility described in *Obtaining Results File Output in Abaqus/Explicit* and *Abaqus/Standard and Abaqus/Explicit Execution*.

Input File Usage: Use the first option in conjunction with one or more of the subsequent options to request output to the Abaqus/Explicit selected results file:

**FILE OUTPUT*
**EL FILE*
**ENERGY FILE*
**NODE FILE*

Output Frequency

You can control the frequency of all Abaqus/Explicit results file output for a particular step by specifying the number of intervals during the step at which file output will be written, *n*. The data are always written at the start and end of each step in which a results file output request is active. The times at which the results are written are referred to as time marks.

If the specified number of intervals is 10, Abaqus/Explicit will write results 11 times: the values at the beginning of the step and at the end of 10 equal time intervals throughout the step. The specified number of intervals must be a positive integer.

By default, results will be written at the increment ending immediately after each time mark. Alternatively, you can choose to have the time increment size adjusted so that an increment will end exactly at each of the time marks calculated by dividing the step into *n* equal intervals.

Input File Usage: Use the following option to request results at the increments ending immediately after each time interval:

**FILE OUTPUT*, NUMBER INTERVAL=*n*, TIME MARKS=NO

Use the following option to request results at the exact time intervals:

**FILE OUTPUT*, NUMBER INTERVAL=*n*, TIME MARKS=YES

Requesting Output in Multiple Steps

Output requests apply to the step in which they are defined and to all subsequent steps until they are respecified.

One exception occurs when the step type changes from general to linear perturbation (available only in Abaqus/Standard). Output requests defined in general steps apply only to subsequent general steps; output requests defined in linear perturbation steps apply only to subsequent consecutive linear perturbation steps. In other words, output defined in a general step is independent of output defined in a linear perturbation step. Propagation between linear perturbation steps occurs only for consecutive linear perturbation steps. If a general analysis step occurs between perturbation steps, output defined in the first perturbation step will not propagate to the next perturbation step. In addition, section output requests are not propagated among linear perturbation steps in Abaqus/Standard.

Element Output

You can output element variables (stresses, strains, section forces, element energies, etc.) for a particular step to the Abaqus/Standard data (`.dat`) file, the Abaqus/Standard results (`.fil`) file, or the Abaqus/Explicit selected results (`.sel`) file. The output requests can be repeated as often as necessary within a step to define output for different types of element variables, different element sets, etc. The same element (or element set) can appear in several output requests.

In general, element output requests remain in effect for subsequent steps unless they are redefined; the appearance of a single element output request in a step removes all element output requests from a previous step. See *About Output* for a discussion of requesting output in multiple general analysis steps or linear perturbation steps.

In Abaqus/Explicit the element output is written to the selected results (`.sel`) file, which must be converted to the results (`.fil`) file as explained above.

Input File Usage: Use the following option to output element variables to the Abaqus/Standard data file:

**EL PRINT*

Use the following option to output element variables to the Abaqus/Standard results file or the Abaqus/Explicit selected results file:

**EL FILE*

Selecting the Element Output Variables

The following types of element variables are recognized for the purpose of defining output:

- "Element integration point" variables are associated with the integration points at which the material calculations are performed (for example, components of stress and strain). For beams and pipes defined in Abaqus/Standard with a general beam section, integration point variables are available only if the output section points were specified for the section (see *Using a General Beam Section to Define the Section Behavior*). For first-order heat transfer elements the integration points are located at the corners of the element in heat capacitance calculations.
- "Element section point" variables are associated with the cross-section of a beam, pipe, or a shell (for example, bending moments and membrane forces on the section).
- "Whole element" variables are attributes of an entire element (for example, the total energy content of the element).
- "Whole element set" variables are attributes of an entire element set (for example, the current coordinates of the center of mass); these variables are available only in Abaqus/Standard.

The element variables that can be written to the data and results files are defined in *Abaqus/Standard Output Variable Identifiers* and *Abaqus/Explicit Output Variable Identifiers*.

Abaqus/Standard allows only complete sets of basic variables (for example, all of the stress or strain components) to be written to the results file. Individual variables (such as a particular stress component) cannot be selected and must be obtained by postprocessing. Abaqus/Standard element variables can be written to the data and results files at the integration points, at the centroid, averaged at the nodes, or extrapolated to the nodes.

In Abaqus/Explicit the complete stress or strain tensors can be written to the selected results file, or individual scalar variables such as equivalent plastic strain can be written. Abaqus/Explicit writes element variables to the results file only at the integration points where they are calculated.

Selecting the Elements for Which Output Is Required

You can specify the element set for which output is being requested. If you do not specify an element set, the output will be printed for all elements and, in Abaqus/Explicit, for all rebars in the model. In Abaqus/Standard output requests for rebars are governed separately, as discussed below.

Input File Usage: Use either of the following options:

**EL PRINT*, ELSET=*element_set_name*
**EL FILE*, ELSET=*element_set_name*

Specifying the Section Point in Beams, Pipes, Shells, and Layered Solid Elements

For beams, pipes, shells, or layered solid elements in Abaqus/Standard output is provided at the default section points listed in *Abaqus Elements Guide*. You can specify nondefault output points.

In Abaqus/Explicit output is always provided at all section points for beam, pipe, and shell element output requests.

Input File Usage: Use either of the following options in Abaqus/Standard:

**EL PRINT*
list of output points
**EL FILE*
list of output points

Requesting Output for Rebars in a Reinforced Model

In Abaqus/Standard you can request output for rebars (*Defining Reinforcement*). If you do not explicitly request rebar output in an Abaqus/Standard model with rebars, the element output requests govern the output for the matrix material only (except for section forces, where the forces in the rebar are included in the force calculation). You can request output for a particular rebar. If you do not specify the name of a rebar, output will be given for all rebars in the specified element set (or in the whole model, if you have not specified an element set).

In beam and continuum elements in Abaqus/Standard rebar output can be obtained at the integration points only. In shell, membrane, and surface elements rebar output is available at the integration points and at the element's centroid.

In Abaqus/Explicit output for the rebars in the specified element set (or the whole model, if you have not specified an element set) is always included for element output requests.

Input File Usage: Use either of the following options in Abaqus/Standard:

**EL PRINT*, REBAR=*rebar_name*
**EL FILE*, REBAR=*rebar_name*

Selecting the Position of Element Integration and Section Point Output in Abaqus/Standard

In Abaqus/Standard integration point variables and section variables can be written to the data and results files in four different positions. By default, output is provided at the integration points.

Obtaining Element Output at the Integration Points

By default, the variables are output at the integration points where they are calculated. (You can obtain the position of the integration points by using output variable COORD—see *Abaqus/Standard Output Variable Identifiers*.)

Input File Usage: Use either of the following options:

EL PRINT, POSITION=INTEGRATION POINTS
EL FILE, POSITION=INTEGRATION POINTS

Obtaining Element Output at the Centroid of Each Element

You can choose to output the variables at the centroid of each element (the centroid of the reference surface of a shell element or the midpoint between the end nodes of a beam or a pipe element). Centroidal values are obtained by interpolation of the integration point values if the integration scheme for the element does not include a centroidal integration point.

Input File Usage: Use either of the following options:

EL PRINT, POSITION=CENTROIDAL
EL FILE, POSITION=CENTROIDAL

Obtaining Element Output Averaged at the Nodes

You can choose to extrapolate the variables to the nodes, then average them over all of the elements in the set that contribute to each node. For derived variables, such as the principal stress, Abaqus/Standard will first average the extrapolated tensor components over all of the elements connected to the node to obtain unique components at each node, then calculate the derived value based on the averaged components.

By default, Abaqus/Standard partitions the elements in the model into averaging regions. The partitioning is based upon the structure of the elements: element type, number of section points, type of material, single layer or composite, etc. Partitioning is not based upon the values of element properties (such as thickness), material orientations, or material constants. Averaging will occur only over elements that contribute to a node and belong to the same averaging region.

In some situations you may want the averaging regions to take into account the values of element properties. For example, since variables may be discontinuous between elements with different material constants, you may not want elements with different property definitions included in the same averaging region. In such cases you can force Abaqus/Standard to take into account values of element properties by setting the Abaqus environment parameter **average_by_section** to `ON`. However, in problems with many section and/or material definitions the default value of `OFF` will, in general, give much better performance than the nondefault value of `ON`.

Input File Usage: Use either of the following options:

EL PRINT, POSITION=AVERAGED AT NODES
EL FILE, POSITION=AVERAGED AT NODES

Obtaining Element Output Extrapolated to the Nodes

You can choose to extrapolate the element integration point variables to the nodes of each element independently, without averaging the results from adjoining elements.

Input File Usage: Use either of the following options:

EL PRINT, POSITION=NODES
EL FILE, POSITION=NODES

Extrapolation and Interpolation of Element Output Variables

The shape functions of the element are used for purposes of extrapolation and interpolation of output variables. Extrapolated values are generally not as accurate as the values calculated at the integration points in the areas of high stress gradients, particularly in the case of modified triangles and tetrahedra. Therefore, adequately detailed meshing is necessary around nodes where accurate nodal values of such element results are needed. If a cylindrical or spherical coordinate system is defined for the element (see *Orientations*), the orientation at each integration point may be different. When the values at the integration points are extrapolated to the nodes, the difference in the orientation is not taken into account; therefore, if the orientation varies significantly over the elements connected to a node, the extrapolated values will not be very accurate. If the material orientation undergoes significant spatial variation in a region of the model where the material behavior is truly anisotropic, a finer mesh is required to obtain accurate results even at the integration points. In that situation once the overall solution has converged with respect to the mesh density, the interpolation or extrapolation away from the integration points can also be assumed to be reasonably accurate. Element output for second-order elements with one collapsed side in two dimensions or one collapsed face in three dimensions should not be extrapolated to the nodes.

In a coupled temperature-displacement and a coupled thermal-electrical-structural analysis nodal temperatures (variable NT11) are more accurate than temperatures at the integration point (variable TEMP) extrapolated to the nodes.

For derived variables, such as the Mises equivalent stress, the components are first extrapolated or interpolated, then the derived value is calculated from the extrapolated or interpolated components. However, in linear mode-based dynamic analysis procedures where values are obtained as nonlinear combinations of modal response magnitudes (*Random Response Analysis* and *Response Spectrum Analysis*), the nonlinear combinations are first calculated at the integration points. These derived values are extrapolated to the nodes or interpolated to the centroid.

Requesting Summaries in the Abaqus/Standard Data File

By default in Abaqus/Standard, summaries of element variables are printed in the data file. A summary of the maximum and minimum values is printed at the end of each column in an output table. The locations of the maximum and minimum values are also printed. You can choose to suppress this summary.

Input File Usage: **EL PRINT*, SUMMARY=YES *or* NO

Requesting Totals in the Abaqus/Standard Data File

In Abaqus/Standard you can print the sum (total) of each column in an output table to the data file. Totals can be used, for example, to obtain a sum of all the energies in a set of elements. By default, these totals are suppressed.

Input File Usage: **EL PRINT*, TOTALS=YES *or* NO

Controlling the Frequency of Output

In Abaqus/Standard you can control the frequency of element output by specifying the output frequency in increments. Unless a frequency of zero is specified to suppress output, the variables will always be output at the last increment of the step.

In Abaqus/Explicit the frequency of element output is controlled as described in *Output Frequency* above.

Input File Usage: Use either of the following options in Abaqus/Standard:

**EL PRINT*, FREQUENCY=*n*
**EL FILE*, FREQUENCY=*n*

Specifying the Directions for Element Output

For components of stress, strain, and similar material variables, 1, 2, and 3 refer to the directions in an orthogonal coordinate system. If a local orientation is not defined for the element, the stress/strain components are in the default directions defined by the convention given in *Conventions*: global directions for solid elements; surface directions for shell, membrane, and gasket elements; and axial and transverse directions for beam and pipe elements.

If a local orientation is associated with the element, the element output variable components are in the local directions defined by the orientation (see *Orientations*). In Abaqus/Standard you can request that the local directions be written to the results file if component output is requested for any variable (see *Output of Local Directions to the Results File* below). In Abaqus/Explicit the local directions will always be written to the results file when tensor output is requested for any element variable. The local directions are written automatically to the output database file from both Abaqus/Standard and Abaqus/Explicit.

In large-displacement problems the local directions defined in the reference configuration are rotated into the current configuration by the average material rotation. See *State storage* for details.

Controlling the Output during Eigenvalue Extraction

You can control element output during natural frequency extraction (*Natural Frequency Extraction*), complex eigenvalue extraction (*Complex Eigenvalue Extraction*), and eigenvalue buckling analysis (*Eigenvalue Buckling Prediction*) by specifying the first and last mode numbers for which output is required. By default, the first mode number is 1 and the last mode number is *N*, where *N* is the number of modes extracted. If you specify the first mode number, the default value for the last mode number is *M*, where *M* is the value specified for the first mode number.

Input File Usage: Use either of the following options:

**EL PRINT*, MODE=*m*, LAST MODE=*n*
**EL FILE*, MODE=*m*, LAST MODE=*n*

Abaqus/Standard Data File Format

In Abaqus/Standard the printed output of variables is arranged in tables in the data file. For element variables, each row of a table corresponds to a particular location: an element, a node, a section point within an element, or an integration point. The rows that will appear in a particular table are defined by choosing an element set and, possibly, locations within each element in the set.

Each table is defined by a data line of the element output request, which specifies the variables to appear in that table. There is no limit to the number of tables that can be defined. The first columns of a table define the location—the element or node number, integration point number, etc. You choose which data will appear in the remaining columns; up to 9 variables (columns) can appear in a table. For example, output variables S and E cannot be requested on the same data line in a three-dimensional analysis because that would produce 12 columns of output. If all of the entries in a row are zero, the row is not printed.

Each table can contain only one type of output variable (whole element, section, or integration point); one type of element; and only one type of section definition. If an element output request to the data file includes more than one type of output variable, element, or section definition, Abaqus/Standard will split the output automatically into the necessary number of individual tables. All of the tables defined by the first data line of the output request will be printed, then all of the tables defined by the second data line, etc.

Results File Format

An element header record (the type 1 record described in *Results File*) is created for each line of requests for each integration point and section point in an element. In addition to the element header record, a direction record (record type 85) can be written in Abaqus/Standard when complete stress or strain tensor output is requested (see below). In Abaqus/Explicit a direction record is always written when complete stress or strain tensor output is requested.

For Abaqus/Standard file output requests with multiple variables, it is advantageous to specify as many variables as possible on each data line of the element output request (up to 16). By keeping the number of lines of requests to a minimum, extra type 1 and type 85 records are avoided and the size of the results file may be reduced substantially. This is not an issue in Abaqus/Explicit. Element variables must be of the same "type" (element integration point variable; element section variable; whole element variable; etc.) to be entered on a single line—see *About Output*. In Abaqus/Standard if all results in a file output record are zero, the record is not written to the results file.

Output of Local Directions to the Results File

By default, in Abaqus/Standard the local coordinate directions are not written to the results file. If component output is requested, you can write the local coordinate directions to the results file. A direction record of type 85 will be written following the type 1 record.

In Abaqus/Explicit the local coordinate directions are always written to the selected results file as a direction record of type 85 when complete stress or strain tensor output is requested.

Tensor component output is given in the local coordinate system, which may be inherent to the element (as is the case in shells and membranes) or user-defined (*Orientations*).

For shell elements a direction record is written for every material point in the section for which component output is requested, and a separate direction record is written for section forces and section strains. For geometrically nonlinear analysis in Abaqus/Standard the record contains the current, updated directions, except for small-strain shells and gasket elements, for which the original directions are given. For three-dimensional beams, direction output is written only if section output has been requested.

Direction output is not provided for trusses, two-dimensional beams, two-dimensional gasket elements, axisymmetric shells, axisymmetric membranes, axisymmetric gasket elements, or for values averaged at nodes. In addition, it is not provided for GK*xx*N-type gasket elements, which have no membrane or transverse shear deformation.

Input File Usage: Use the following option in Abaqus/Standard:

**EL FILE*, DIRECTIONS=YES

Default Element Output

If you do not specify an element output request to the results file in a step (or in any previous step of the analysis), no element output will be written to the results file; similarly, if you do not specify an element output request to the data file (available only in Abaqus/Standard) in a step (or in any previous step of the analysis), no element output will be written to the data file.

Node Output

You can output nodal variables (displacements, reaction forces, etc.) for a particular step to the Abaqus/Standard data (`.dat`) file, the Abaqus/Standard results (`.fil`) file, or the Abaqus/Explicit selected results (`.sel`) file. The output requests can be repeated as often as necessary within a step to define output for different node sets. The same node (or node set) can appear in several output requests.

In general, nodal output requests remain in effect for subsequent steps unless they are redefined; the appearance of a single nodal output request in a step removes all nodal output requests from a previous step. See *About Output* for a discussion of requesting output in multiple general analysis steps or linear perturbation steps.

In Abaqus/Explicit the nodal output is written to the selected results (`.sel`) file, which must be converted to the results (`.fil`) file as explained above.

Input File Usage: Use the following option to output nodal variables to the Abaqus/Standard data file:

**NODE PRINT*

Use the following option to output nodal variables to the Abaqus/Standard results file or the Abaqus/Explicit selected results file:

**NODE FILE*

Selecting the Nodal Output Variables

The nodal variables that can be written to the data and results files are defined in the "Nodal variables" portion of *Abaqus/Standard Output Variable Identifiers* and *Abaqus/Explicit Output Variable Identifiers*.

Abaqus allows only complete sets of basic variables (for example, all of the displacement components) to be written to the results file. Individual variables (such as a particular displacement component) cannot be selected and must be obtained by postprocessing.

Selecting the Nodes for Which Output Is Required

You can specify the node set for which output is being requested. If you do not specify a node set, the output will be printed for all nodes in the model.

Input File Usage: Use either of the following options:

**NODE PRINT*, NSET=*node_set_name*
**NODE FILE*, NSET=*node_set_name*

Requesting Summaries in the Abaqus/Standard Data File

By default in Abaqus/Standard, summaries of nodal variables are printed in the data file. A summary of the maximum and minimum values is printed at the end of each column in an output table. The locations of the maximum and minimum values are also printed. You can choose to suppress this summary.

Input File Usage: **NODE PRINT*, SUMMARY=YES *or* NO

Requesting Totals in the Abaqus/Standard Data File

In Abaqus/Standard you can print the sum (total) of each column in an output table to the data file. Totals can be used, for example, to sum reaction forces at the nodes. By default, these totals are suppressed.

Input File Usage: **NODE PRINT*, TOTALS=YES *or* NO

Controlling the Frequency of Output

In Abaqus/Standard you can control the frequency of nodal output by specifying the output frequency in increments. Unless a frequency of zero is specified to suppress output, the variables will always be output at the last increment of the step.

In Abaqus/Explicit the frequency of nodal output is controlled as described in *Output Frequency* above.

Input File Usage: Use either of the following options in Abaqus/Standard:

**NODE PRINT*, FREQUENCY=*n*
**NODE FILE*, FREQUENCY=*n*

Specifying the Directions for Nodal Output

For nodal variables 1, 2, and 3 refer to the global directions *X*, *Y*, and *Z*, respectively. For axisymmetric elements 1 and 2 refer to the global directions *r* and *z*.

In Abaqus/Standard components of nodal variables such as reaction forces are output in the global directions unless a local coordinate system has been defined at a node (see *Transformed Coordinate Systems*). In this case you can specify whether output is desired in global or local directions. The local directions defined by the nodal transformation cannot be written to the results file.

The data in the Abaqus/Explicit selected results file are always output in the global directions, even if a local coordinate system has been defined at a node.

Obtaining Nodal Output in the Global Directions

In Abaqus/Standard you can request vector-valued nodal variables in the global directions, which is the default for nodal output requests to the results file since most postprocessors assume that components are given in the global system.

Input File Usage: Use either of the following options:

**NODE PRINT*, GLOBAL=YES
**NODE FILE*, GLOBAL=YES

Obtaining Nodal Output in the Local Directions Defined by Nodal Transformations

In Abaqus/Standard you can request vector-valued nodal variables in the local directions defined by nodal transformations, which is the default for nodal output requests to the data file.

Input File Usage: Use either of the following options:

**NODE PRINT*, GLOBAL=NO
**NODE FILE*, GLOBAL=NO

Controlling the Output during Eigenvalue Extraction

You can control nodal output during natural frequency extraction, complex eigenvalue extraction, and eigenvalue buckling analysis by specifying the first and last mode numbers for which output is required, as described above for element output.

Input File Usage: Use either of the following options:

**NODE PRINT*, MODE=*m*, LAST MODE=*n*
**NODE FILE*, MODE=*m*, LAST MODE=*n*

Abaqus/Standard Data File Format

In Abaqus/Standard the printed output of variables is arranged in tables by node set in the data file. For nodal variables each row of a table corresponds to an individual node.

Each table is defined by a data line of the nodal output request, which specifies the variables to appear in that table. There is no limit to the number of tables that can be defined. The first column of each table is the node number. You choose the variables to appear in the remaining columns; up to nine variables (columns) can appear in a table. If all of the entries in a row are zero, the row is not printed. Displacement, velocity, and acceleration components less than a relative tolerance (equal to 100 times the machine precision times the current maximum value in the model) are treated as zero.

Results File Format

There is no header or direction record for nodes, so it makes little difference whether items are requested on a single line or multiple lines. In Abaqus/Standard if all results in a record are zero, the record is not written to the results file.

Default Nodal Output

If you do not specify a nodal output request to the results file in a step (or in any previous step of the analysis), no nodal output will be written to the results file; similarly if you do not specify a nodal output request to the data file (available only in Abaqus/Standard) in a step (or in any previous step of the analysis), no nodal output will be written to the data file.

Total Energy Output

You can output summaries of the energy content of the model to the Abaqus/Standard data (`.dat`) file, the Abaqus/Standard results (`.fil`) file, or the Abaqus/Explicit selected results (`.sel`) file. Energy output requests are not available for the following procedures:

- *Eigenvalue Buckling Prediction*
- *Natural Frequency Extraction*
- *Complex Eigenvalue Extraction*

Energy output requests remain in effect for subsequent steps. Detailed energy density output is available by using element output requests (see *Element Output*).

In Abaqus/Explicit the energy output is written to the selected results (`.sel`) file, which must be converted to the results (`.fil`) file as explained above.

Input File Usage: Use the following option to output summaries of the energy content to the Abaqus/Standard data file:

**ENERGY PRINT*

Use the following option to output summaries of the energy content to the Abaqus/Standard results file or the Abaqus/Explicit selected results file:

**ENERGY FILE*

External Work Calculation due to Concentrated Follower Forces

Abaqus/Standard may generate inaccurate external work (ALLWK) in the presence of a concentrated follower load that rotates with time (see *Specifying Concentrated Follower Forces*). This problem may occur in both static and implicit dynamic analyses and may result in an inaccurate total energy (ETOTAL) history output. Other results (displacements, stresses, strains, etc.) are not affected. The inaccuracy is due to the fact that the increment of work is calculated using the direction of the concentrated load at the end of the increment instead of using an average load over the increment.

Energy Computation Accuracy

Energy terms may not be computed consistently. Some of the energy terms are integrated using the trapezoidal rule (for example, elastic energy in Abaqus/Standard, which has second-order accuracy for smooth problems). Other terms, such as contact frictional dissipation, are computed using the backward difference method, which is only first-order accurate. The total energy balance may not be constant in time due to such discrepancies, especially in the presence of discontinuities such as contact impact.

Selecting the Energy Output Variables

When energy output is requested, all of the total energy quantities listed in *Abaqus/Standard Output Variable Identifiers* or *Abaqus/Explicit Output Variable Identifiers* are output; the variables cannot be selected individually.

Selecting the Element Set for Which Total Energy Output Is Required

In Abaqus/Standard you can specify the element set for which total energy output is being requested. In this case the energies are summed for all the elements in the specified set. You cannot specify an element set for the following procedures:

- *Transient Modal Dynamic Analysis*
- *Mode-Based Steady-State Dynamic Analysis*
- *Response Spectrum Analysis*
- *Random Response Analysis*

If you do not specify an element set, the total energies for the whole model will be output. If total energy output for both the whole model and for different element sets is desired, the energy output requests must be repeated; once without a specified element set to request energy output for the whole model and once for each specified element set.

In Abaqus/Explicit you cannot specify selected element sets for an energy output request; the total energies for the whole model will always be output.

Input File Usage: Use one of the following options in Abaqus/Standard:

**ENERGY PRINT*, ELSET=*element_set_name*
**ENERGY FILE*, ELSET=*element_set_name*

Controlling the Frequency of Output

In Abaqus/Standard you can control the frequency of energy output by specifying the output frequency in increments. Unless a frequency of zero is specified to suppress output, the variables will always be output at the last increment of the step.

In Abaqus/Explicit the frequency of energy output is controlled as described in *Output Frequency* above.

Input File Usage: Use either of the following options in Abaqus/Standard:

**ENERGY PRINT*, FREQUENCY=*n*
**ENERGY FILE*, FREQUENCY=*n*

Default Energy Output

Energy output requests must be included for total energy output to be written to the data and results files; no default output is provided.

Modal Output from Abaqus/Standard

You can output generalized coordinate (modal amplitude and phase) values during modal dynamic procedures (see *About Dynamic Analysis Procedures* for an overview of the modal dynamic procedures available in Abaqus/Standard) to the data (`.dat`) file or results (`.fil`) file.

You can also request that eigenvalues be written to the results file during *Eigenvalue Buckling Prediction* or *Natural Frequency Extraction*. The eigenvalues are always written to the results file when element or nodal output to the results file is requested; however, modal output requests allow you to write the eigenvalues to the results file without requesting any additional output.

Input File Usage: Use the following option to output modal variables to the Abaqus/Standard data file:

**MODAL PRINT*

Use the following option to output modal variables to the Abaqus/Standard results file:

**MODAL FILE*

Selecting the Modal Output Variables

The modal variables that can be written to the data and results files are defined in the “Modal variables” portion of *Abaqus/Standard Output Variable Identifiers*.

Controlling the Frequency of Output

You can control the frequency of modal output by specifying the output frequency in increments. Unless a frequency of zero is specified to suppress output, the variables will always be output at the last increment of the step.

Input File Usage: Use either of the following options:

**MODAL PRINT*, FREQUENCY=*n*
**MODAL FILE*, FREQUENCY=*n*

Default Modal Output

Modal output requests must be included for modal results to be written to the data and results files; no default output is provided.

Surface Output from Abaqus/Standard

In Abaqus/Standard you can write variables associated with surfaces in contact, coupled temperature-displacement, coupled thermal-electrical-structural, coupled thermal-electrical, and crack propagation problems to the data and results files. The output requests can be repeated as often as necessary within a step to define output for different contact pairs and different types of surface variables.

See *Cavity Radiation in Abaqus/Standard* for information on requesting output of surface variables associated with cavity radiation.

Use element output requests (see *Element Output*) to obtain data and results file output for contact elements (such as slide line elements; see *Slide Line Contact Elements*).

Selecting the Surface Output Variables

The following types of surface variables are recognized for the purpose of defining output:

- "Secondary node" variables are associated with the integration points at which the material calculations are performed (for example, the contact stress).
- "Whole surface" variables are attributes of an entire secondary surface (for example, the total force due to contact pressure).

The surface variables that can be written to the data and results files are listed in the "Surface variables" portion of *Abaqus/Standard Output Variable Identifiers*.

Selecting the Contact Pairs for Which Output Is Required

You can select the main and secondary surfaces for which output is required, and you can specify a subset of secondary nodes for output in addition to the main and secondary surfaces or independently. If no surfaces or secondary nodes are specified, surface variables are written for all the contact pairs in the model. If you specify the secondary surface but not the main surface, output is given for all contact pairs that involve the specified secondary surface.

Input File Usage: Use either of the following options:

CONTACT PRINT, MAIN=*main*, SECONDARY=*secondary*, NSET=*node_set*
CONTACT FILE, MAIN=*main*, SECONDARY=*secondary*, NSET=*node_set*

Requesting Summaries in the Data File

By default, summaries of surface variables are printed in the data file. A summary of the maximum and minimum values is printed at the end of each column in an output table. The locations of the maximum and minimum values are also printed. You can choose to suppress this summary.

Input File Usage: *CONTACT PRINT*, SUMMARY=YES *or* NO

Requesting Totals in the Data File

You can print the sum (total) of each column in an output table to the data file. By default, these totals are suppressed.

Input File Usage: *CONTACT PRINT*, TOTALS=YES *or* NO

Controlling the Frequency of Output

You can control the frequency of surface output by specifying the output frequency in increments. Unless a frequency of zero is specified to suppress output, the variables will always be output at the last increment of the step.

Input File Usage: Use either of the following options:

CONTACT PRINT, FREQUENCY=*n*
CONTACT FILE, FREQUENCY=*n*

Default Surface Output

Surface output requests must be included for surface variables associated with contact pairs to be written to the data and results files; no default output is provided.

If a surface output request is defined without any specified output variables, the following variables will be written to the data and results files by default:

- For contact analysis, contact pressure (CPRESS), frictional shear stresses (CSHEAR), contact opening (COPEN), and relative tangential motions (CSLIP); see *About Contact Pairs in Abaqus/Standard*.
- For heat transfer analysis, heat flux per unit area (HFL), heat flux (HFLA), time integrated HFL (HTL), and time integrated HFLA (HTLA); see *Thermal Contact Properties*.
- For coupled thermal-electrical analysis, HFL, HFLA, HTL, HTLA, electrical current per unit area (ECD), electrical current (ECDA), time integrated ECD (ECDT), and time integrated ECDA (ECDTA); see *Electrical Contact Properties*.
- For coupled pore fluid-mechanical analysis, CPRESS, CSHEAR, COPEN, CSLIP, pore fluid volume flux per unit area (PFL), pore fluid volume flux (PFLA), time integrated PFL (PTL), and time integrated PFLA (PTLA); see *Pore Fluid Contact Properties*.
- For crack propagation analysis, there are no default output quantities; bond failure quantities must be requested explicitly; see *Crack Propagation Analysis*.

Data File Format

Printed output of variables is arranged in tables. Each table is defined by a data line of the surface output request, which specifies the variables to appear in that table. Each table can contain only one type of output variable (secondary node or whole surface). For example, output variables CSTRESS and CFN cannot be requested on the same data line. For the secondary node type of output, each row of a table corresponds to a node on the secondary surface. The rows that will appear in a particular table will be limited to the node set specified in the output request. The first column of each table defines the location (the node number). The remaining columns contain variables such as contact pressure, frictional shear stresses, contact opening, and relative tangential (slip) motions. For the whole surface type of output, each row of a table corresponds to an entire secondary surface. If all of the variables in a row of a table are zero, the row is not printed.

If a contact output request refers to more than one contact pair, a separate table will be generated for each contact pair. All of the tables defined by the first data line of the output request will be printed, then all of the tables defined by the second line, etc.

Results File Format

A contact output request record (the type 1503 record described in *Results File*) is created for each output request. For the secondary node type of output, this record is followed by several node header records, each of which contains a node on the secondary surface. Each node header record is followed by records that contain output variables. The output will be limited to the node set specified in the output request. For the whole surface type of output, the type 1503 record is followed by only one type 1504 node header record with a node number zero. The node header record is followed by records containing the requested output variables.

If a contact output request refers to more than one contact pair, a separate contact output request record is generated for each contact pair.

Section Output from Abaqus/Standard

In Abaqus/Standard you can output accumulated quantities associated with user-defined sections (see *Abaqus/Standard Output Variable Identifiers*) for a particular step to the data or results file. This facility provides "free body diagram" output, allowing analyses of force flow through a redundant structure. The output requests can be repeated as often as necessary within a step to define output for different sections and different section output variables. You can assign a label to each output request that will be used to identify the output for the section. Section output is not available for eigenfrequency extraction, eigenvalue buckling prediction, complex eigenfrequency extraction, or linear dynamics procedures or in procedures using multiple load cases.

Defining the Surface Section

Section output requests are available only for sections defined using element-based surfaces (see *Element-Based Surface Definition*). Consequently, the sections must be defined using faces of continuum elements although other types of elements (beams, membranes, shells, springs, dashpots, etc.) can be attached to the section.

Calculation of accumulated quantities on the section (such as the total force) involves nodal quantities associated with elements on one side of the section only. Therefore, the surface definition should use elements only from one side of the section (the "base elements," as defined in *Prescribed Assembly Loads*), thus precisely identifying the side from which accumulated quantities are computed.

Since the section usually cuts through the mesh in a typical section output request, automatic generation of the surface cannot be used. Specifying the element faces gives exact control over which element faces form the surface, which is essential when defining a cross-section through a solid body.

You must specify the name of the surface for which output is being requested.

Surfaces that are defined in a restart analysis can be used only for section output requests. The newly defined surface cannot be used for any other purpose (such as a contact pair or pre-tension section definition).

Input File Usage: Use either of the following options:

**SECTION PRINT*, NAME=*section_name*, SURFACE=*surface_name*
**SECTION FILE*, NAME=*section_name*, SURFACE=*surface_name*

Example

For example, the following input illustrates a typical section output request to the data file:

```
*HEADING
Section print example
…
*SURFACE, NAME=surface_name
Data lines that specify the elements and their associated faces to define the
surface section
…
*STEP
…
*SECTION PRINT, NAME=section_name,
SURFACE=surface_name, …
…
*END STEP
```

Alternatively, if additional section output requests are needed after the analysis is completed, a restart analysis can be performed to request more output as shown in the following input:

```
*RESTART, READ, …
…
*SURFACE, NAME=surface_name
Data lines that specify the elements and their associated faces to define the
surface section
…
*STEP
…
*SECTION PRINT, NAME=section_name,
SURFACE=surface_name, …
…
*END STEP
```

Selecting the Coordinate System in Which Output Is Desired

You can specify the choice of coordinate system in which the section output is desired. By default, the components of vector quantities associated with the section are obtained with respect to the global system of coordinates. Alternatively, you can specify that output is desired in a local system as defined below.

Input File Usage: Use either of the following options:

**SECTION PRINT*, NAME=*section_name*, SURFACE=*surface_name*, AXES=GLOBAL or LOCAL
**SECTION FILE*, NAME=*section_name*, SURFACE=*surface_name*, AXES=GLOBAL or LOCAL

Defining a Coordinate System Local to the Surface Section

You can allow Abaqus/Standard to define the local system, or you can specify it directly.

Default Local System

The default local system is particularly useful when the section is flat or almost flat. While it can also be used in the case when the defined surface is curved, the default local system may be irrelevant for such problems.

The default system is defined by a straight line in two-dimensional and axisymmetric cases or by a plane in three-dimensional cases, fitted (in a least square sense) through the nodes belonging to the section. The anchor point (origin) of the local system is the centroid of the projection of the surface on the fitted line or plane. The local directions are given by the normal (1-direction) and the tangent direction (the 2-direction in two-dimensional and axisymmetric cases) or the tangent directions (the 2- and 3-directions in three-dimensional cases) to the fitted line or plane. When several straight lines or planes can be fit equally well between the nodes defining the section (for example, a closed circular or spherical surface), the original local directions will be parallel to the global axes.

The positive local 1-direction is selected such that it will form an acute angle with the average normal direction to the section, computed by averaging the positive normals to the element faces defining the section. If the average normal direction is zero (a closed surface), the 1-direction will form an acute angle with the global x-axis. If in two-dimensional or axisymmetric cases the 1-direction is within 0.1° of being normal to the global x-axis, it will form an acute angle with the global y-axis. In three-dimensional cases if the 1-direction is within 0.1° of being normal to the global X–Y plane, it will form an acute angle with the global z-axis.

In two-dimensional and axisymmetric cases the local 2-direction is obtained by rotating the local 1-direction counterclockwise by 90° about the anchor point. For three-dimensional situations the tangent directions of the surface are defined using the Abaqus conventions for local directions on surfaces in space (see *Conventions*).

Input File Usage: Use either of the following options to use the default local coordinate system:

**SECTION PRINT*, NAME=*section_name*, SURFACE=*surface_name*, AXES=LOCAL
**SECTION FILE*, NAME=*section_name*, SURFACE=*surface_name*, AXES=LOCAL

User-Specified Local System

A user-specified local system is defined by specifying the origin and the directions of the axes. You can specify the origin (anchor point) by giving a node number or by specifying the coordinates of the anchor point.

In two-dimensional and axisymmetric cases the local 2-direction is defined by specifying either a predefined node number or the coordinates of a point (point a) on the local 2-direction. The local 1-direction is then obtained by rotating the local 2-axis clockwise by 90° about the anchor point (see *Figure 1*). If node numbers are used to define the anchor point or the local directions, they must be connected to the mesh.

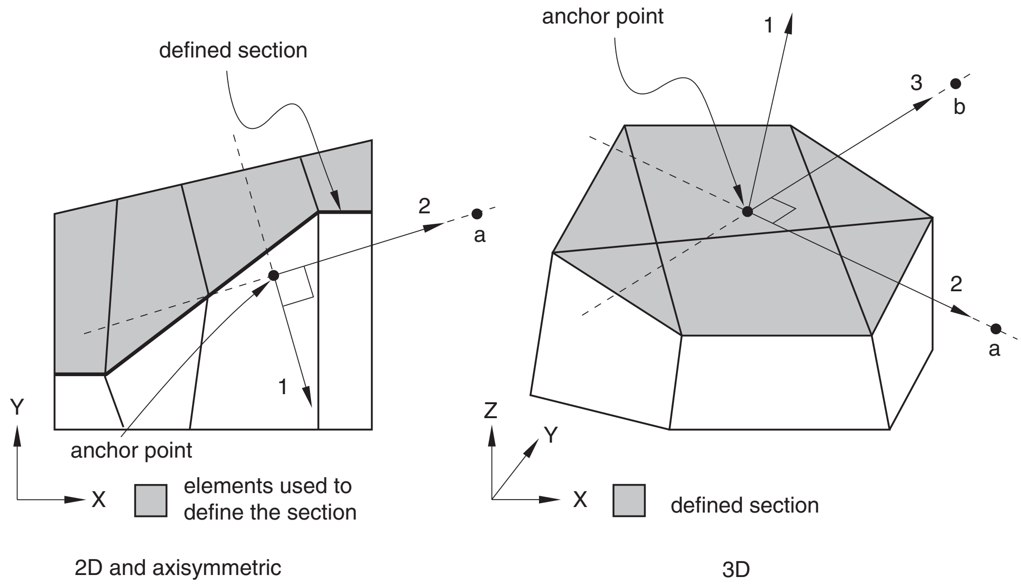


Figure 1: User-defined local coordinate system.

In three-dimensional cases either two predefined nodes or the coordinates of two points can be used to specify the local directions. A rectangular Cartesian coordinate system is then defined by its origin (the anchor point) and these two points. The first point (point *a*) must lie on the local 2-direction, and the second (point *b*) must be in the local 2–3 plane on the side of the local 3-direction. Although it is not necessary, it is intuitive to select the second point such that it is on or near the local 3-direction (see *Figure 1*).

If you do not specify the anchor point of the local system, it is taken to be the centroid of the projection of the surface on the fitted line or plane. If you do not specify the directions of the axes, the local system will be anchored at the specified anchor point and its axes will be parallel to the default axes of the projected surface. If neither the anchor point nor the directions are defined, the default local system will be used.

In large-deformation analyses the surface section may rotate significantly during the deformation. By default, when output is requested in a local coordinate system, the system rotates with the average rigid body motion of the elements used to define the surface section (i.e., the local system and the output are updated during the analysis). The anchor point and local directions must then be specified relative to the undeformed configuration. You can choose to obtain vector output in the original local coordinate system instead. This choice is irrelevant in steps in which geometric nonlinearities are not considered.

Input File Usage: Use either of the following options to specify the local coordinate system directly:

**SECTION PRINT*, NAME=*section_name*, SURFACE=*surface_name*,
AXES=LOCAL, UPDATE=YES or NO
anchor point definition
axes definition
**SECTION FILE*, NAME=*section_name*, SURFACE=*surface_name*,
AXES=LOCAL, UPDATE=YES or NO
anchor point definition
axes definition

Controlling the Frequency of Output

You can control the frequency of section output by specifying the output frequency in increments. Unless a frequency of zero is specified to suppress output, the variables will always be output at the last increment of the step.

Input File Usage: Use either of the following options:

**SECTION PRINT*, NAME=*section_name*, SURFACE=*surface_name*, FREQUENCY=*n*
**SECTION FILE*, NAME=*section_name*, SURFACE=*surface_name*, FREQUENCY=*n*

Data File Format

Printed output is arranged in tables. The first line of the table contains the name of the requested output variable (see *Abaqus/Standard Output Variable Identifiers*), and the second line contains the corresponding value. If a section output request is defined without any specified output variables, all appropriate variables associated with the current analysis type are output.

If several section output requests to the data file are encountered in one particular step, separate tables will be created for each request. Each table has a header denoting the name of the section and the name of the surface used. In addition, if the output is requested in a local coordinate system, the global coordinates of the anchor point and the cosine directions of the local axes are output.

Results File Format

Several section output records (record numbers 1580–1591 in *Results File*) are output for each section output request to the results file. The actual collection of records to be written to the results file depends on the number of valid output requests. If a section output request is defined without any specified output variables, all records relevant to the current analysis type are stored in the results file.

Vector Output in the Section

Vector output associated with section output requests consists of the total force (SOF), the total moment (SOM), and the center of forces (SOCF). Output variable SOF is computed as a vector sum of the stress-based (internal) nodal forces of the nodes in the surface.

Output variable SOM is computed with respect to the origin of the coordinate system considered. Thus, if the output is requested in the global coordinate system, the total moment is computed about the global origin; if the output is requested in a local coordinate system, the moment is computed about the current anchor point of the local system. The coordinates of the current anchor point may change during the analysis if the local coordinate system is updated. Output variables SOF and SOM are both reported in the coordinate system considered.

The center of forces SOCF is computed as the closest point to the centroid of the section through which the total force SOF acts. SOCF is always reported in the global coordinate system. If the total force vector is equal to zero, the centroid of the section is reported as the center of forces SOCF.

The total moment vector, SOM, will not necessarily equal the cross product of the center of force vector, SOCF, and total force vector, SOF. Forces acting on two different points of the section may have components acting in opposite directions, such that these force components generate a net moment but not a net force; therefore, the total moment may not arise entirely from the resultant force.

Scalar Output in the Section

Scalar output associated with a section output request consists of the area of the defined section (SOAREA), the total heat flux (SOH) in heat transfer analysis, the total current (SOE) in electrical analysis, the total mass flow (SOD) in mass diffusion analysis, and the total pore fluid volume flux (SOP) in couple pore fluid diffusion-stress analysis. These output variables are computed as the algebraic sum of the scalar internal nodal fluxes (work-conjugate to the associated primary solution variables) of the nodes in the surface. For example, in heat transfer analysis the total heat flux (SOH) is the sum of the NFLUX values at the nodes on the surfaces.

Limitations When Using Section Output Requests

Section output requests are subject to the following limitations:

- Section output requests are available only for sections defined by an element-based surface. Thus, they can be used only for sections along faces of continuum elements.
- When defining the section, elements on only one side of the section must be used. Abaqus/Standard identifies all elements attached to the surface on this side and computes the section output variables as in a free-body diagram.
- The defined section must cut completely through the mesh, form a closed surface, or be on the exterior of the body. *Figure 2* presents some typical cases of valid surfaces. If the section cuts only partially through the mesh, a valid free-body diagram cannot be isolated (see *Figure 3*) and incorrect answers may be computed. Abaqus/Standard will attempt to identify the invalid cases and will issue error or warning messages.

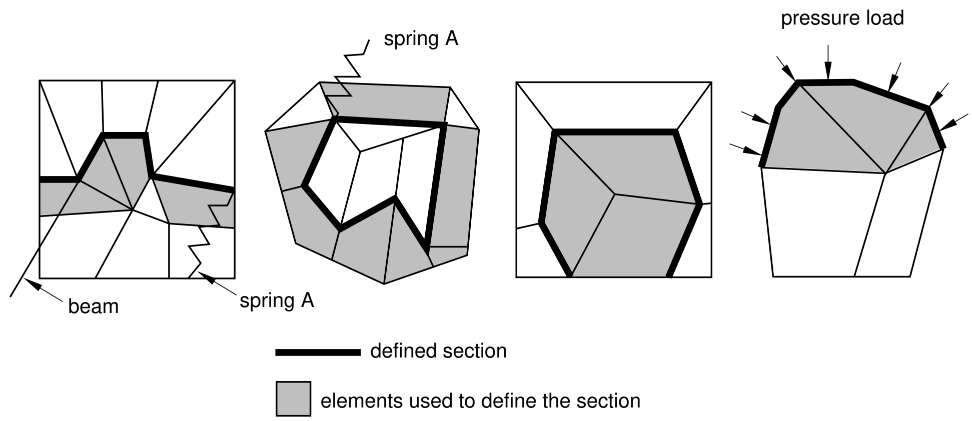


Figure 2: Valid section definitions.

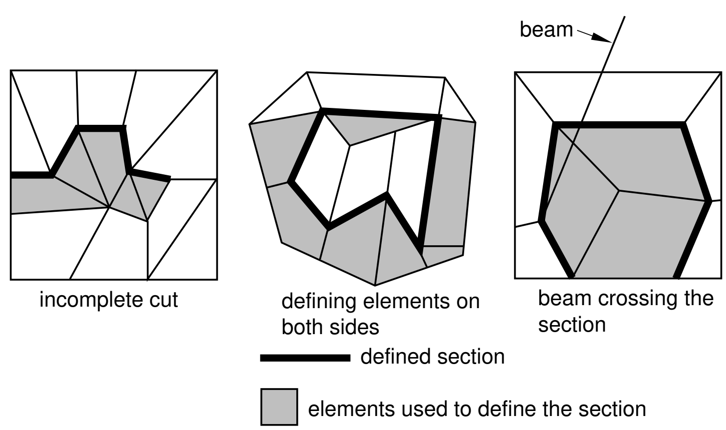


Figure 3: Invalid section definitions.

- Elements attached to the section can be on either side of the surface but must not cross the defined section. *Figure 3* presents a few invalid cases. In most cases Abaqus/Standard will successfully identify elements that cross the surface, and warning messages will be issued. The elements will then not be considered in the calculation of the section variables.
- For section output purposes, Abaqus/Standard will ignore the elements attached to the section for which it cannot establish whether they belong to one side or the other of the section (e.g., SPRING1 elements).
- Section output requests cannot be specified within a substructure.
- Section output requests cannot be specified in random response analyses.
- The total force and the total moment in the section are computed based only on the stresses (internal forces) in the identified elements. Thus, inaccurate results may be obtained if distributed body loads are present in these elements since their effect on the total force in the section is not included. Common examples are the inertial loading in dynamic analyses, gravity loads, distributed body forces, and centrifugal loads. In these cases the total force in the section may depend on the choice of elements used to define the section as illustrated in *Figure 4*(a). Assuming that gravity loading is the only active load, the element stresses will be different in the two elements. Hence, if the same section is defined first using element 1 and then using element 2, different answers for the total force will be obtained. In a similar way the effects of any distributed body fluxes (heat, electrical, etc.) prescribed in the identified elements are not included.

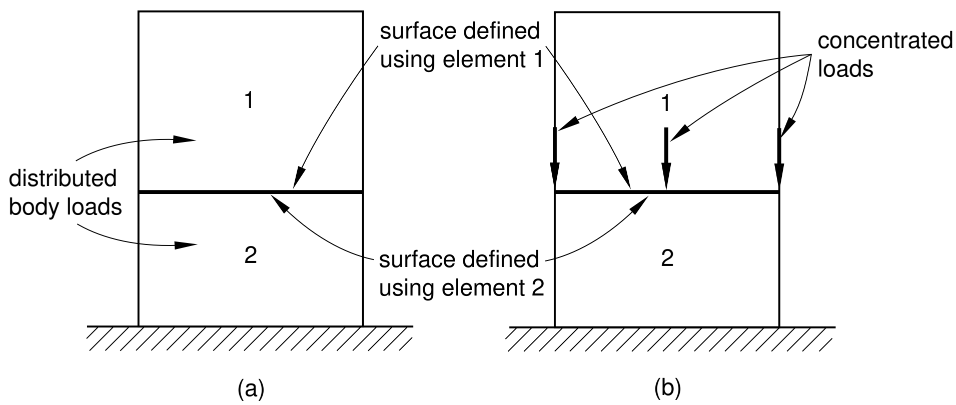


Figure 4: Total force in the section.

- Depending on which side of the surface is used to define the section, different answers will be obtained in analyses similar to the case illustrated in *Figure 4*(b). Assuming a static analysis with the concentrated loads shown in the figure being the only active loads, a zero total force is reported if the section is defined using element 1 and a nonzero force equal to the sum of the concentrated loads is obtained if the section is defined using element 2.

Output to the Output Database

Output variables are available for:

- element integration points, element section points, whole elements, and element sets;
- surfaces in Abaqus/Explicit;
- integrated output sections in Abaqus/Explicit and Abaqus/Standard;
- nodes; and
- the whole model.

All the output variables are defined in *Using Abaqus/Standard Output Variable Identifiers* and *Using Abaqus/Explicit Output Variable Identifiers*.

Model information and analysis results are stored in terms of an assembly of part instances (see *Assembly Definition*).

See the *Abaqus Scripting User's Guide* for a description of how to use the Abaqus Scripting Interface or C++ to access an output database.

In this section:

Requesting Output to the Output Database

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

Three types of information are stored in the output database in Abaqus/Standard and Abaqus/Explicit: "field" output, "history" output, and diagnostic information.

Field output and history output are controlled by output database requests as described in this section. A subset of the diagnostic information that is written to the message file for Abaqus/Standard analyses and to the status and message files for Abaqus/Explicit analyses is included in the output database.

- Field output is intended for infrequent requests for a large portion of the model and can be used to generate contour plots, animations, symbol plots, *X–Y* plots, and displaced shape plots in Abaqus/CAE. Only complete sets of basic variables (for example, all the stress or strain components) can be requested as field output.
- History output is intended for relatively frequent output requests for small portions of the model and is displayed in *X–Y* data plots in Abaqus/CAE. Individual variables (such as a particular stress component) can be requested.
- Diagnostic information in Abaqus/Standard and Abaqus/Explicit is intended to provide analysis warning and/or error information as well as convergence information for use in Abaqus/CAE.

Output database requests can be repeated as often as necessary within a step to produce both field and history output at multiple frequencies.

Requesting Field Output

Contact surface output, element output, nodal output, and radiation output are available as field output in Abaqus/Standard and Abaqus/Explicit.

Input File Usage: Use the first option in conjunction with one or more of the subsequent options to request field output to the output database:

**OUTPUT*, FIELD
**CONTACT OUTPUT*
**ELEMENT OUTPUT*
**NODE OUTPUT*
**RADIATION OUTPUT*

These options are discussed in detail below.

Abaqus/CAE Usage: Step module: field output request editor

Requesting History Output

Contact surface output, element output, energy output, integrated output, time incrementation output, modal output, nodal output, and radiation output are available as history output in Abaqus/Standard and Abaqus/Explicit.

Requesting large amounts of history output (more than 1000 output requests) may cause performance to degrade in Abaqus/Standard and will cause performance to degrade in Abaqus/Explicit. For vector- or tensor-valued output variables each component is considered to be a single request. In the case of element variables history output will be generated at each integration point. For example, requesting history output of the tensor variable S (stress) for a C3D10M element will generate 24 history output requests: (6 components) × (4 integration points). When requesting history output of vector- and tensor-valued variables, it is recommended that individual components be selected where available.

Input File Usage: Use the first option in conjunction with one or more of the subsequent options to request history output to the output database:

**OUTPUT*, HISTORY
**CONTACT OUTPUT*
**ELEMENT OUTPUT*
**ENERGY OUTPUT*
**INTEGRATED OUTPUT*
**INCREMENTATION OUTPUT*
**MODAL OUTPUT*
**NODE OUTPUT*
**RADIATION OUTPUT*

These options are discussed in detail below.

Abaqus/CAE Usage: Step module: history output request editor

Requesting Diagnostic Information

By default, a subset of the diagnostic information that is written to the message file for Abaqus/Standard analyses and to the status and message files for Abaqus/Explicit analyses is also written to the output database. You can use the Visualization module of Abaqus/CAE to view this diagnostic information interactively, highlighting problematic areas on a view of the model and using them to resolve errors and warnings in the analysis. For more information, see *The Message File in Abaqus/Standard and Abaqus/Explicit*, and *Viewing diagnostic output*.

Input File Usage: Use the following option to write diagnostic information to the output database:

**OUTPUT*, DIAGNOSTICS=YES

Use the following option to exclude diagnostic information:

**OUTPUT*, DIAGNOSTICS=NO

Abaqus/CAE Usage: You cannot exclude diagnostic information from the output database from within Abaqus/CAE. Use the following option to view the saved diagnostic information:

Visualization module: **Tools**->**Job Diagnostics**

Controlling the Frequency of Output to the Output Database

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

The frequency of output to the output database is controlled differently in Abaqus/Standard and Abaqus/Explicit. Control of the output frequency in Abaqus/Explicit depends upon whether field or history output was selected.

Controlling the Output Frequency in Abaqus/Standard

Abaqus/Standard provides several options for controlling the output frequency, depending on whether the analysis is in the time domain (e.g., general statics), frequency domain (e.g., steady state dynamics), or mode domain (e.g., natural frequency extraction). These options can be used to reduce the amount of output written and hence improve performance and disk space use as compared to the default output.

History output in Abaqus/Standard is buffered and is written to disk only after every 10 increments of history data output or when a step has completed. Therefore, history results may not be available immediately for postprocessing.

Default Output Frequency

If you do not specify the output frequency, field and history output will be written at every increment of the analysis for all procedure types except dynamic and modal dynamic analyses for which output will be written every 10 increments.

Controlling Output Frequency in a Frequency Domain Analysis

In frequency domain procedures, you only can control the frequency of output by specifying the frequency of output in increments. The data will be written at this frequency as well as at the end of each step of the analysis. Specify an output frequency of zero to suppress output.

Input File Usage: **OUTPUT*, FREQUENCY=*n*

Abaqus/CAE Usage: Step module: field or history output request editor: **Frequency**: **Every *n* increments**: *n*

Controlling Output Frequency in a Mode Domain Analysis

In an eigenvalue extraction or eigenvalue buckling analysis, you can select the modes at which output is desired. If you do not specify a list of modes, output is produced for all of the modes.

Input File Usage: **OUTPUT*, FIELD, MODE LIST

Abaqus/CAE Usage: Step module: field output request editor: **Frequency**: **Specify modes**: *list of modes*

Controlling Output Frequency in a Time Domain Analysis

In time domain analyses, you can control the frequency of output by specifying the output frequency in terms of increments, the number of intervals during the step, the size of regular time intervals throughout the step, or time points throughout the step. The different options are described in more detail below.

Whichever option is chosen, the output will always be written at the zero-increment and last increment of the analysis and, for a low-cycle fatigue analysis, at the end of each cycle. The zero-increment output represents the initial conditions for the current analysis step and is essential for sequential thermal-stress analyses and analyses involving submodeling, for which a complete solution history (including the solution state at the beginning of the step) is needed to ensure proper interpolation in time. The zero-increment state is written at the beginning of the step, before the solution of the incremental nonlinear finite-element equations for the step commences, and is therefore in general not an equilibrium solution. Particular examples where the solution is not in equilibrium include the first step of an analysis in which an initial stress state is defined and when loads or boundary condition changes are discontinuous between steps.

Usually, the zero-increment output in any step corresponds to the base state, which is the state of the model at the end of the last general step. The exception to this is modal transient dynamic analysis, where the zero-increment output represents the linear perturbation response at time zero.

By default, when convergence difficulties are encountered in a general step, output is written for the last converged increment. To recover the requested results variables for this last converged increment, a new attempt is performed. There is no message written to the status file or the message file to show this additional attempt. In the output database file you will see an extra attempt and an additional frame. If the previous increment was written to the output database and convergence difficulties are encountered during the current increment, the last converged increment is still written to the output database, which will result in a duplicate output frame at the end of the analysis.

Time Domain Analysis: Specifying Output Frequency in Increments

You can specify how frequently you want output in terms of increments. Specify an output frequency of zero to suppress output.

Input File Usage: **OUTPUT*, FREQUENCY=*n*

Abaqus/CAE Usage: Step module: field or history output request editor: **Frequency**: **Every *n* increments**: *n*

Time Domain Analysis: Specifying Output Frequency in Number of Intervals

You can specify the output frequency in number of intervals, *n*. The specified number of intervals must be a positive integer.

By default, Abaqus/Standard adjusts the time increment (in some cases Abaqus/Standard might violate the minimum time increment specified) to ensure that data are written at the exact times calculated by dividing the step into *n* equal intervals. Alternatively, you can specify that the data be written immediately after each time mark. In this case no adjustment of the time increment is necessary.

Input File Usage: Use the following option to request results at the exact time intervals:

**OUTPUT*, NUMBER INTERVAL=*n*, TIME MARKS=YES

Use the following option to request results at the increments ending immediately after each time interval:

**OUTPUT*, NUMBER INTERVAL=*n*, TIME MARKS=NO

Abaqus/CAE Usage: Use the following option to request results at the exact time intervals:

Step module: field or history output request editor: **Frequency**: **Evenly spaced time intervals**, **Interval**: *n*, **Timing**: **Output at exact times**

Use the following option to request results at the increments ending immediately after each time interval:

Step module: field or history output request editor: **Frequency**: **Evenly spaced time intervals**, **Interval**: *n*, **Timing**: **Output at approximate times**

Time Domain Analysis: Specifying Output Frequency in Regular Time Interval Size

You can write the results at specified regular intervals throughout the step as well as at the end of the step.

By default, Abaqus/Standard will adjust the time increment (in some cases Abaqus/Standard might violate the minimum time increment specified) to ensure that data will be written at the exact times, as defined by multiples of the time interval, *t*. Alternatively, the data can be written immediately after each time mark. In this case no adjustment of the time increment is necessary.

Input File Usage: Use the following option to request results at the exact time intervals:

**OUTPUT*, TIME INTERVAL=*t* , TIME MARKS=YES

Use the following option to request results at the increments ending immediately after each time interval:

**OUTPUT*, TIME INTERVAL=*t* , TIME MARKS=NO

Abaqus/CAE Usage: Use the following option to request results at the exact time intervals:

Step module: field or history output request editor: **Frequency**: **Every *x* units of time**: *t*, **Timing**: **Output at exact times**

Use the following option to request results at the increments ending immediately after each time interval:

Step module: field or history output request editor: **Frequency**: **Every *x* units of time**: *t*, **Timing**: **Output at approximate times**

Time Domain Analysis: Specifying Output Frequency in Time Points

You can write the results at specified time points throughout the step.

By default, Abaqus/Standard adjusts the time increment (in some cases Abaqus/Standard might violate the minimum time increment specified) to ensure that data are written at the exact time points specified. Alternatively, you can specify that the data be written immediately after each time point. In this case no adjustment of the time increment is necessary.

Input File Usage: Use the following options to request results at the exact time points:

**TIME POINTS*, NAME=*time points name*
**OUTPUT*, TIME POINTS=*time points name*, TIME MARKS=YES

Use the following options to request results at the increments ending immediately after each time point:

**TIME POINTS*, NAME=*time points name*
**OUTPUT*, TIME POINTS=*time points name*, TIME MARKS=NO

Abaqus/CAE Usage: Use the following option to request results at the exact time points:

Step module: field or history output request editor: **From time points**, **Name**: *time points name*, **Timing**: **Output at exact times**

Use the following option to request results at the increments ending immediately after each time point:

Step module: field or history output request editor: **From time points**, **Name**: *time points name*, **Timing**: **Output at approximate times**

Time Domain Analysis: Time Incrementation

If the output frequency is specified at exact times and in terms of the number of intervals, in regular time intervals, or in time points, Abaqus/Standard adjusts the time increments to ensure that data are written at the exact time points. In some cases Abaqus may use a time increment smaller than the minimum time increment allowed in the step in the increment directly before a time point. However, Abaqus will not violate the minimum time increment allowed for consolidation, transient mass diffusion, transient heat transfer, transient couple thermal-electrical, transient coupled temperature-displacement, and transient coupled thermal-electrical-structural analyses. For these procedures if a time increment smaller than the minimum time increment is required, Abaqus will use the minimum time increment allowed in the step and will write output data at the first increment after the time point.

When the output frequency is specified at exact times and in terms of the number of intervals, in regular time intervals, or in time points, the number of increments necessary to complete the analysis might increase, which might adversely affect performance.

Controlling the Output Frequency for Field Output in Abaqus/Explicit

Field output data are always written at the start and end of each step in which the output request is active. In addition, you can specify the output frequency in terms of the number of intervals during the step, the size of regular time intervals throughout the step, or time points throughout the step. The times at which the results are written are referred to as time marks.

Specifying Field Output Frequency in Number of Intervals

You can specify the output frequency in number of intervals, n. The specified number of intervals must be a positive integer. For example, if the specified number of intervals is 10, Abaqus/Explicit will write field data 11 times: the values at the beginning of the step and at the end of 10 equal time intervals throughout the step.

By default, field data will be written at the increment ending immediately after each time mark. Alternatively, when you specify the output frequency in number of intervals, you can choose to have the time increment size adjusted so that an increment will end exactly at each of the time marks calculated by dividing the step into *n* equal intervals.

Input File Usage: Use the following option to request results at the increments ending immediately after each time interval:

**OUTPUT*, FIELD, NUMBER INTERVAL=*n*, TIME MARKS=NO

Use the following option to request results at the exact time intervals:

**OUTPUT*, FIELD, NUMBER INTERVAL=*n*, TIME MARKS=YES

Abaqus/CAE Usage: Use the following option to request results at the increments ending immediately after each time interval:

Step module: field output request editor: **Frequency**: **Evenly spaced time intervals**, **Interval**: *n*, **Timing**: **Output at approximate times**

Use the following option to request results at the exact time intervals:

Step module: field output request editor: **Frequency**: **Evenly spaced time intervals**, **Interval**: *n*, **Timing**: **Output at exact times**

Specifying Field Output Frequency in Regular Time Interval Size

Alternatively, you can write the results at specified regular intervals throughout the step as well as at the beginning and end of the step. The time increment size will not be adjusted to meet the specified time marks; results will be written at the increment ending immediately after each time mark, as defined by multiples of the time interval, *t*.

Input File Usage: **OUTPUT*, FIELD, TIME INTERVAL=*t*

Abaqus/CAE Usage: Step module: field output request editor: **Frequency**: **Every *x* units of time**: *t*

Specifying Field Output Frequency in Time Points

You can write the results at specified time points throughout the step. Regular time intervals between time points are not required; you can specify any desired time points at which the field output is to be written.

Input File Usage: Use the following option to request results at the exact time points:

**TIME POINTS*, NAME=*time points name*
**OUTPUT*, FIELD, TIME POINTS=*time points name*, TIME MARKS=YES

Use the following option to request results at the increments ending immediately after each time point:

**TIME POINTS*, NAME=*time points name*
**OUTPUT*, FIELD, TIME POINTS=*time points name*, TIME MARKS=NO

Abaqus/CAE Usage: Use the following option to request results at the exact time points:

Step module: field output request editor: **Frequency**: **From time points**, **Name**: *time points name*, **Timing**: **Output at exact times**

Use the following option to request results at the increments ending immediately after each time point:

Step module: field output request editor: **Frequency**: **From time points**, **Name**: *time points name*, **Timing**: **Output at approximate times**

Default Field Output

If you do not specify the output frequency (in either number of intervals, time interval size, or time points), field output will be written at 20 equally spaced intervals throughout the step.

Controlling the Output Frequency for History Output in Abaqus/Explicit

If history output is selected, you can specify the output frequency in terms of either increments or regular intervals throughout the step.

Specifying History Output Frequency in Increments

You can specify the output frequency in increments. The data will be written at this frequency as well as at the end of each step of the analysis.

Input File Usage: **OUTPUT*, HISTORY, FREQUENCY=*n*

Abaqus/CAE Usage: Step module: history output request editor: **Frequency**: **Every *n* time increments**: *n*

Specifying History Output Frequency in Regular Time Interval Size

Alternatively, you can write the results at specified regular intervals throughout the step as well as at the end of the step. The time increment size will not be adjusted to meet the specified time marks; results will be written at the increment ending immediately after each time mark, as defined by multiples of the time interval, *t*.

Input File Usage: **OUTPUT*, HISTORY, TIME INTERVAL=*t*

Abaqus/CAE Usage: Step module: history output request editor: **Frequency**: **Every *x* units of time**: *t*

Default History Output

If you do not specify the output frequency (in either increments or time interval size), history output will be written at 200 equally spaced intervals throughout the step.

Controlling the Precision of Element and Nodal Output

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- *Abaqus/Standard and Abaqus/Explicit Execution*
- *The job editor*

Overview

You can control the precision of element and nodal output.

You can control the precision of element output for an Abaqus/Standard analysis. You can control the precision of nodal output for an Abaqus/Standard or Abaqus/Explicit analysis.

Input File Usage: Use the following command line option to request single-precision element output in Abaqus/Standard or to request single-precision nodal output:

abaqusjob=*job-name* **output_precision**=`single`

Use the following command line option to request double-precision element output in Abaqus/Standard or double-precision nodal output:

abaqusjob=*job-name* **output_precision**=`full`

Abaqus/CAE Usage: Job module: job editor: **Precision**: **Nodal output precision: Single** or **Full**

You cannot control the precision of element output in Abaqus/CAE.

Requesting Output in Multiple Steps

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

Output requests apply to the step in which they are defined and to all subsequent steps until they are respecified.

The only exception occurs when the step type changes from general to linear perturbation (available only in Abaqus/Standard). Output requests defined in general steps apply only to subsequent general steps; output requests defined in linear perturbation steps apply only to subsequent consecutive linear perturbation steps. In other words, output defined in a general step is independent of output defined in a linear perturbation step. Propagation between linear perturbation steps occurs only for consecutive linear perturbation steps. If a general analysis step occurs between perturbation steps, output defined in the first perturbation step will not propagate to the next perturbation step.

In any given step you can add or selectively replace the output requests that are continued from previous steps. Alternatively, you can discontinue all requests from previous steps and request a completely new set of output. The preselected field variables and preselected history output variables (see *Preselected Output Requests*) are requested by default for the first step of an analysis; you can modify this request as in any other step.

Specifying New Output Requests

By default, all output requests defined in previous steps are removed when new requests are defined, regardless of the type of output request being defined. In other words, a new field output request in a step removes all field and history output requests defined in previous steps.

Because all existing output requests are removed when a new request is defined in a step, all output requests within the same step are treated as new (i.e., additional output requests or replacement output requests are treated as equivalent to new output requests).

Input File Usage: Use one of the following options to remove all existing output requests and to specify new requests:

**OUTPUT*, FIELD, OP=NEW
**OUTPUT*, HISTORY, OP=NEW

Abaqus/CAE Usage: Step module: **Create Field Output Request** or **Create History Output Request**

Abaqus/CAE automatically respecifies all previously defined output requests when you create a new request.

Specifying Additional Output Requests

Alternatively, you can specify additional output requests without removing all default and previously defined output requests.

Input File Usage: Use one of the following options to specify additional output requests without removing all default and previously defined output requests:

**OUTPUT*, FIELD, OP=ADD
**OUTPUT*, HISTORY, OP=ADD

Abaqus/CAE Usage: Step module: **Create Field Output Request** or **Create History Output Request**

Abaqus/CAE automatically respecifies all previously defined output requests when you create a new request.

Replacing or Removing an Output Request

You can replace an output request of the same type (e.g., field or history) and frequency with a new request. No other previously defined requests will be affected.

You cannot replace an output request to change its frequency. If no matching request is found, the request specified is simply added to the step.

To remove a previously defined request, you can replace the output request without specifying any new output variables.

Input File Usage: Use one of the following options to replace an output request with a new request:

**OUTPUT*, FIELD, OP=REPLACE
**OUTPUT*, HISTORY, OP=REPLACE

Abaqus/CAE Usage: Step module: **Field Output Requests Manager** or **History Output Requests Manager**: **Edit** or **Delete**

Suppressing Output Requests Defined in Previous Steps

To suppress completely all output requests that have been defined in previous steps, you can specify an output frequency of 0.

Preselected Output Requests

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

There are two ways to define output variable requests quickly and easily. Both methods are available for field and history output requests and for the individual output requests used for requesting specific variable types (for example, nodal, element).
The use of these methods with individual output requests for specific variable types is explained in detail later in this section.

Requesting Procedure-Specific Preselected Output Requests

You can activate a procedure-specific set of commonly requested output variables. See *Table 1* for a list of procedure types and their accompanying preselected variables. The variables written to the output database may change if the procedure type changes between steps.

Table 1: List of preselected variables for various procedure types.

Procedure type	Preselected element variables (field)	Preselected nodal and surface variables (field)	Preselected energy variables (history)
Annealing	none	none	none
Complex frequency extraction	none	U	none
Coupled pore fluid diffusion/stress	S, E, VOIDR, SAT, POR	U, RF, CF, PFL, PFLA, PTL, PTLA, TPFL, TPTL	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Coupled thermal-electric	HFL, EPG, GRADT	NT, RFL, EPOT	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Direct cyclic	S, E, PE, PEEQ, PEMAG	U, RF, CF	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN,

Procedure type	Preselected element variables (field)	Preselected nodal and surface variables (field)	Preselected energy variables (history)
			ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Direct-integration implicit dynamic (with an output frequency of 10)	S, E, PE, PEEQ, PEMAG	U, V, A, RF, CF, CSTRESS, CDISP	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Direct-solution steady-state dynamic	S, E	U, V, A, RF, CF	ALLKE, ALLSE, ALLVD, ALLWK
Eigenfrequency extraction	none	U	none
Eigenvalue buckling prediction	none	U	none
Explicit dynamic	S, LE, PE, PEEQ, EVF, SVAVG, PEVAVG, PEEQVAVG	U, V, A, RF, CSTRESS	ALLKE, ALLSE, ALLWK, ALLPD, ALLCD, ALLVD, ALLDMD, ALLAE, ALLIE, ALLFD, ETOTAL
Fatigue crack growth	S, E	U, RF, CSTRESS	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Fully coupled thermal-electrical-structural in Abaqus/Standard	S, E, PE, PEEQ, PEMAG, HFL, EPG, GRADT	U, RF, CF, NT, RFL, CSTRESS, CDISP, EPOT	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Fully coupled thermal-stress in Abaqus/Standard	S, E, PE, PEEQ, PEMAG, HFL, GRADT	U, RF, CF, NT, RFL, CSTRESS, CDISP	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Fully coupled thermal-stress in Abaqus/Explicit	S, LE, PE, PEEQ, HFL	U, V, A, RF, CSTRESS, NT, RFL	ALLKE, ALLSE, ALLWK, ALLPD, ALLCD, ALLVD, ALLDMD, ALLAE, ALLIE, ALLFD, ALLIHE, ALLHF, ETOTAL

Procedure type	Preselected element variables (field)	Preselected nodal and surface variables (field)	Preselected energy variables (history)
Geostatic stress field	S, E, POR, SAT, VOIDR	U, RF, CF, CSTRESS, CDISP	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Heat transfer	HFL, GRADT	NT, RFL	none
Linear static perturbation	S, E	U, RF, CF	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Mass diffusion	CONC, MFL	NNC, RFL	none
Modal dynamic (with an output frequency of 10)	S, E	U, V, A, RF, CF	ALLAE, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
SIM-based modal dynamic	none	none	none
Quasi-static	S, E, PE, PEEQ, PEMAG, CE, CEEQ, CEMAG	U, RF, CF, CSTRESS, CDISP	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Random response	S, E	U, V, A	none
Response spectrum	S, E	U, RF, CF	ALLKE, ALLSE, ALLWK
Static	S, E, PE, PEEQ, PEMAG	U, RF, CF, CSTRESS, CDISP	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD, ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Steady-state dynamic	S, E	U, V, A, RF, CF	ALLKE, ALLSE, ALLWK
SIM-based steady-state dynamic	none	none	none
Steady-state transport	S, E	U, RF, CF, CSTRESS, CDISP	ALLAE, ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLDMD,

Procedure type	Preselected element variables (field)	Preselected nodal and surface variables (field)	Preselected energy variables (history)
			ALLWK, ALLKL, ALLQB, ALLEE, ALLJD, ALLSD, ETOTAL
Subspace-based steady-state dynamic	S, E	U, V, A, RF, CF	ALLKE, ALLSE, ALLVD, ALLWK

If you request preselected field or history output and request additional output variables using individual output requests for specific variable types, the variables requested will be appended to the variables contained in the preselected list.

For geometrically nonlinear analysis in Abaqus/Standard, E is not available for output and LE is output by default. For linear perturbation analyses and geometrically linear analyses in Abaqus/Standard, LE and NE strain output requests yield the same output as E. For geometrically linear analysis in Abaqus/Explicit, LE is output.

Abaqus may omit some preselected variables from the analysis results. Abaqus omits preselected output variables if they are not applicable for the element type used to mesh the model or if other factors make the variables unsuitable for the analysis.

Input File Usage: Use one of the following options:

**OUTPUT*, FIELD, VARIABLE=PRESELECT
**OUTPUT*, HISTORY, VARIABLE=PRESELECT

Abaqus/CAE Usage: Step module: field or history output request editor: **Preselected defaults**

Requesting All Variables Applicable to the Current Procedure and Material Type

You can request all variables applicable to the current procedure and material type. Any individual output requests for specific variable types are ignored in this case.

Input File Usage: Use one of the following options:

**OUTPUT*, FIELD, VARIABLE=ALL
**OUTPUT*, HISTORY, VARIABLE=ALL

Abaqus/CAE Usage: Step module: field or history output request editor: **All**

Writing Default Output to the Output Database

Products: Abaqus/Standard Abaqus/Explicit

References:

- *About Output*
- **OUTPUT*

Overview

If no output database requests are specified, the preselected field and history output variables are written automatically to the output database.

In Abaqus/Standard the default variables are written at every increment for both field and history output for all procedure types except dynamic and modal dynamic analyses; the default frequency for field and history output for these procedure types is every 10 increments. In Abaqus/Explicit the default variables are written at 20 intervals for field output and 200 intervals for history output.

Turning off Default Output

You can turn these defaults off by using the **odb_output_by_default** environment file parameter; see *Environment File Settings* for details. Furthermore, specifying new output database requests in a step (see *Specifying New Output Requests*) overrides the default field and history output requests for that step. For large models the default output to the output database may increase the solution time and required disk space considerably. In such cases you are encouraged to review carefully the relevance of the default output variables for the proposed analysis. A C++ program is available that creates a smaller copy of a large output database by copying data from only selected frames; for more information, see *Decreasing the amount of data in an output database by retaining data at specific frames*.

The **odb_output_by_default** environment file parameter is ignored in a restart analysis. If no output requests are defined in a restart analysis, the output requests are those that propagate from the original analysis.

Abaqus/Explicit Output as a Result of Analysis Termination

When an Abaqus/Explicit analysis encounters a fatal error in an increment, the preselected variables applicable to the current procedure are written automatically to the output database as field data. The analysis will go through an additional increment with a zero time increment size before writing these data.

Writing Element Output to the Output Database

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

You can request that element variables (stresses, strains, section forces, element energies, etc.) be written to the output database.

The output request can be repeated as often as necessary to define output for different types of element variables, different element sets, etc. The same element (or element set) can appear in several output requests. Element output to the output database is not supported for user elements.

Selecting the Element Output Variables

The following types of element variables are recognized for the purpose of defining output:

- "Element integration point" variables are associated with the integration points at which material calculations are performed (for example, components of stress and strain).
- "Element section point" variables are associated with the cross-section of a beam, pipe, or a shell (for example, bending moments and membrane forces on the section).
- "Element face" variables are associated with the faces of a shell or a solid (for example, uniformly distributed pressure load on the face).
- "Whole element" variables are attributes of an entire element (for example, the total energy content of the element).
- "Whole element set" variables are attributes of an entire element set (for example, the current coordinates of the center of mass); these variables are available in Abaqus/Standard and Abaqus/Explicit.

The element variables that can be written to the output database are defined in *Using Abaqus/Standard Output Variable Identifiers* and *Using Abaqus/Explicit Output Variable Identifiers*.

Input File Usage: **ELEMENT OUTPUT*
list of output variables

Abaqus/CAE Usage: Step module: field or history output request editor: **Select from list below**

Selecting Elements for Which Output Is Required

For history output you must specify the element set (or, in Abaqus/Explicit, the tracer set) for which output is being requested. For field output specifying the element set or tracer set is optional; if you do not specify an element set or tracer set, the output will be written for all the elements in the model.

Input File Usage: **ELEMENT OUTPUT*, ELSET=*element_set_name*

Abaqus/CAE Usage: Step module: field or history output request editor: **Domain: Set:** *set_name*

Requesting Field Output for the Exterior Elements in the Model

You can select output on the element set consisting of all the exterior three-dimensional elements in the model. This element set is generated internally by Abaqus.

Input File Usage: **ELEMENT OUTPUT*, EXTERIOR

Abaqus/CAE Usage: Step module: field output request editor: **Domain**: **Whole model**; toggle on **Exterior only**

Specifying the Section Point in Beam, Pipe, Shell, and Layered Solid Elements

For beams, pipes, shells, or layered solids output is provided at the default section points. You can specify nondefault output points.

Input File Usage: **ELEMENT OUTPUT*
list of output points
list of output variables

Abaqus/CAE Usage: Step module: field or history output request editor: **Output at shell, beam, and layered section points: Specify:** *list of output points*

Requesting Output at All Section Points in Beam, Pipe, Shell, and Layered Solid Elements

You can specify that output be provided for all section points in beams, pipes, shells, and layered solids.

Input File Usage: **ELEMENT OUTPUT*, ALLSECTIONPTS

Abaqus/CAE Usage: Requesting output at all section points in beam, pipe, shell, and layered solid elements is not supported in Abaqus/CAE.

Requesting Output for Rebars in a Reinforced Model

You can request output for rebars (*Defining Reinforcement*). If you do not explicitly request rebar output in a model with rebars, the element output requests govern the output for the matrix material only (except for section forces, where the forces in the rebar are included in the force calculation). You can request output for a particular rebar. If you do not specify the name of a rebar, output will be given for all rebars in the specified element set (or in the whole model, if you have not specified an element set).

Rebar output is available only in membrane, shell, or surface elements at the integration points and at the centroid of the element.

Input File Usage: Use the following options:

**OUTPUT*, FIELD
**ELEMENT OUTPUT*, REBAR=*rebar_name*, ELSET=*element_set_name*

**OUTPUT*, HISTORY
**ELEMENT OUTPUT*, REBAR=*rebar_name*, ELSET=*element_set_name*

Abaqus/CAE Usage: Use the following option to request output for rebar in addition to output for the matrix material:

Step module: field or history output request editor: **Output for rebar: Include**

Use the following option to request output only for rebar:

Step module: field or history output request editor: **Output for rebar: Only**

You cannot request output for a particular rebar in Abaqus/CAE; if you request rebar output, it is given for all rebars in the specified output domain.

Selecting the Position of Element Integration Point and Section Point Output

Integration point variables and section variables in Abaqus/Standard can be written as field output to the output database in four different positions: the integration points, the centroid, averaged at nodes, or extrapolated to the nodes. Integration point variables and section variables in Abaqus/Explicit can be written as field output to the output database in three different positions: the integration points, the centroid, or the nodes. By default, output is provided at the integration points.

In most cases Abaqus/Explicit writes only integration point data to the output database. Transferring of results from the integration points to the user-specified position in Abaqus/Explicit is done by the postprocessing calculator. See *The Postprocessing Calculator* for details.

Element history output to the output database is always provided at the integration points.

Obtaining Output at the Integration Points

By default, the variables are output at the integration points where they are calculated. In Abaqus/Standard you can obtain the position of the integration points by using output variable COORD (see *Using Abaqus/Standard Output Variable Identifiers*).

Input File Usage: **ELEMENT OUTPUT*, POSITION=INTEGRATION POINTS

Abaqus/CAE Usage: Step module: field output request editor: **Element output position: Integration points**

Obtaining Output at the Centroid of Each Element

You can choose to output the variables at the centroid of each element (the midpoint between the end nodes of a beam or a pipe element). Centroidal values are obtained by interpolation of the integration point values if the integration scheme for the element does not include a centroidal integration point. Element output of the element centroidal values is not available for recovering results within substructures; for more information, see *Using Substructures*.

Input File Usage: **ELEMENT OUTPUT*, POSITION=CENTROIDAL

Abaqus/CAE Usage: Step module: field output request editor: **Element output position: Centroidal**

Obtaining Element Output Extrapolated to the Nodes

You can choose to extrapolate the element integration point variables to the nodes of each element independently, without averaging the results from adjoining elements. Element output at the element nodes is not available for recovering results within substructures; for more information, see *Using Substructures*.

Input File Usage: **ELEMENT OUTPUT*, POSITION=NODES

Abaqus/CAE Usage: Step module: field output request editor: **Element output position: Nodes**

Obtaining Element Output Averaged at the Nodes in Abaqus/Standard

You can choose to extrapolate the variables to the nodes and to then average them over all of the elements in the set that contribute to each node. For derived variables, such as stress invariants, Abaqus/Standard first averages the extrapolated tensor components over all of the elements connected to the node to obtain unique components at each node and then calculates the derived value based on the averaged components.

By default, Abaqus/Standard partitions the elements in the model into averaging regions. The partitioning is based upon the structure of the elements: element type, number of section points, type of material, single layer or composite, etc. Partitioning is not based upon the values of element properties (such as thickness), material orientations, or material constants. Averaging occurs only over elements that contribute to a node and belong to the same averaging region.

In some situations you may want the averaging regions to take into account the values of element properties. For example, since variables may be discontinuous between elements with different material constants, you may not want elements with different property definitions included in the same averaging region. In such cases you can force Abaqus/Standard to take into account values of element properties by setting the Abaqus environment parameter **average_by_section** to `ON`. However, in problems with many section and/or material definitions the default value of `OFF` will, in general, give much better performance than the nondefault value of `ON`.

The results are available through Abaqus Scripting Interface commands and can be visualized in the Visualization module of Abaqus/CAE (Abaqus/Viewer).

Input File Usage: **ELEMENT OUTPUT*, POSITION=AVERAGED AT NODES

Abaqus/CAE Usage: Step module: field output request editor: **Element output position: Averaged at nodes**

Extrapolation and Interpolation of Element Output Variables

The shape functions of the element are used for purposes of extrapolation and interpolation of output variables. Extrapolated values are generally not as accurate as the values calculated at the integration points in the areas of high stress gradients, particularly in the case of modified triangles and tetrahedra. Therefore, adequately detailed meshing is necessary around nodes where accurate nodal values of such element results are needed. If a cylindrical or spherical coordinate system is defined for the element (see *Orientations*), the orientation at each integration point may be different. When the values at the integration points are extrapolated to the nodes, the difference in the orientation is not taken into account; therefore, if the orientation varies significantly over the elements connected to a node, the extrapolated values are not very accurate. If the material orientation undergoes significant spatial variation in a region of the model where the material behavior is truly anisotropic, a finer mesh is required to obtain accurate results even at the integration points. In that situation once the overall solution has converged with respect to the mesh density, the interpolation or extrapolation away from the integration points can also be assumed to be reasonably accurate. You should also be particularly careful when interpreting output variables extrapolated to the nodes for second-order elements with midside nodes outside the quarter-point region, such as when one edge is collapsed in two dimensions or one face is collapsed in three dimensions.

For derived variables, such as Mises equivalent stress, the components are first extrapolated or interpolated. The derived value is then calculated from the extrapolated or interpolated components. However, in linear mode-based dynamic analysis procedures where derived values are obtained as nonlinear combinations of modal response magnitudes (*Random Response Analysis* and *Response Spectrum Analysis*), the nonlinear combinations are first calculated at the integration points. These derived values are then extrapolated to the nodes or interpolated to the centroid.

Controlling the Output Frequency

The frequency of element output is controlled as described in *Controlling the Frequency of Output to the Output Database*.

Controlling the Precision of Element Output

The precision of element output is controlled as described in *Controlling the Precision of Element and Nodal Output*.

Requesting Preselected Output

You can request the preselected, procedure-specific element output variables described in *Preselected Output Requests*. In this case you can specify additional variables as part of the output request.

Alternatively, you can request all element variables applicable to the current procedure and material type. In this case any additional variables you specify are ignored.

Input File Usage: Use the following option to request the preselected element output variables:

**ELEMENT OUTPUT*, VARIABLE=PRESELECT

Use the following option to request all applicable element output variables:

**ELEMENT OUTPUT*, VARIABLE=ALL

Abaqus/CAE Usage: Step module: field or history output request editor: **Preselected defaults** or **All**

Specifying the Directions for Element Output

For components of stress, strain, and similar material variables 1, 2, and 3 refer to the directions for an orthogonal coordinate system. If a local orientation is not defined for the element, the stress/strain components are in the default directions defined by the convention given in *Orientations*: global directions for solid elements, surface directions for shell and membrane elements, and axial and transverse directions for beam and pipe elements.

By default, the element material directions for element field output are written to the output database. If a local orientation is associated with the element, by default the results displayed in Abaqus/CAE are in the directions defined by the local orientation. These directions can be visualized in Abaqus/CAE by selecting **Plot->Material Orientations** in the Visualization module. You can choose to suppress the direction output to the output database.

Input File Usage: Use the following option to indicate that the element material directions should not be written to the output database:

**ELEMENT OUTPUT*, DIRECTIONS=NO

Abaqus/CAE Usage: Step module: field output request editor: toggle off **Include local coordinate directions when available**

Writing Nodal Output to the Output Database

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

You can output nodal variables (displacements, reaction forces, etc.) to the output database.
The output request can be repeated as often as necessary to define output for different node sets. The same node (or node set) can appear in several output requests.

Selecting the Nodal Output Variables

The nodal variables that can be written to the output database are defined in *Nodal Variables* (Abaqus/Standard) and *Nodal Variables* (Abaqus/Explicit).

Input File Usage: **NODE OUTPUT*
list of output variables

Abaqus/CAE Usage: Step module: field or history output request editor: **Select from list below**

Selecting the Nodes for Which Output Is Required

For history output you must specify the node set (or, in Abaqus/Explicit, the tracer set) for which output is being requested. For field output the specification of the node set or tracer set is optional; if you do not specify a node set or tracer set, the output will be written for all the nodes in the model.

Input File Usage: **NODE OUTPUT*, NSET=*node_set_name*

Abaqus/CAE Usage: Step module: field or history output request editor: **Domain: Set:** *set_name*

Requesting Field Output for the Exterior Nodes in the Model

You can select output on the node set consisting of all the exterior nodes in the model. This node set is generated internally by Abaqus and includes all the nodes that belong to the exterior three-dimensional elements.

Input File Usage: **NODE OUTPUT*, EXTERIOR

Abaqus/CAE Usage: Step module: field output request editor: **Domain**: **Whole model**; toggle on **Exterior only**

Controlling the Output Frequency

The frequency of nodal output is controlled as described in *Controlling the Frequency of Output to the Output Database*.

Controlling the Precision of Nodal Output

The precision of nodal output is controlled as described in *Controlling the Precision of Element and Nodal Output*.

Requesting Preselected Output

You can request the preselected, procedure-specific nodal output variables described in *Preselected Output Requests*. In this case you can specify additional variables as part of the output request.

Alternatively, you can request all nodal variables applicable to the current procedure type. In this case any additional variables you specify are ignored.

Input File Usage: Use the following option to request the preselected nodal output variables:

**NODE OUTPUT*, VARIABLE=PRESELECT

Use the following option to request all applicable nodal output variables:

**NODE OUTPUT*, VARIABLE=ALL

Abaqus/CAE Usage: Step module: field or history output request editor: **Preselected defaults** or **All**

Specifying the Directions for Nodal Field Output

For nodal variables 1, 2, and 3 refer to the global directions X, Y, and Z, respectively. For axisymmetric elements 1 and 2 refer to the global directions r and z. Nodal field results are written to the output database in the global directions. If a local coordinate system is defined at a node (see *Transformed Coordinate Systems*), the local nodal transformations are written to the output database as well. You can apply these transformations to the results in the Visualization module of Abaqus/CAE to view components in the local systems.

Specifying the Directions for Nodal History Output

For nodal variables 1, 2, and 3 refer to the global directions X, Y, and Z, respectively. For axisymmetric elements 1 and 2 refer to the global directions r and z. Nodal history results are written to the output database in the global directions unless a local coordinate system has been defined at a node (see *Transformed Coordinate Systems*). In this case you can specify whether output is desired in global or local directions.

Obtaining Nodal History Output in the Global Directions

You can request vector-valued nodal variables in the global directions, which is the default for nodal history output requests to the output database since most postprocessors assume that components are given in the global system.

Input File Usage: **NODE OUTPUT*, GLOBAL=YES

Abaqus/CAE Usage: Step module: history output request editor: **Domain**: **Set**: toggle on **Use global directions for vector-valued output**

Obtaining Nodal History Output in the Local Directions Defined by Nodal Transformations

You can request vector-valued nodal variables in the local directions defined by nodal transformations.

Input File Usage: **NODE OUTPUT*, GLOBAL=NO

Abaqus/CAE Usage: Step module: history output request editor: **Domain**: **Set**: toggle off **Use global directions for vector-valued output**

Visualizing Boundary Conditions

Boundary conditions can be visualized in the Visualization module of Abaqus/CAE by selecting **View->ODB Display Options**. Click the **Entity Display** tab in the dialog box that appears.

In an Abaqus/Standard analysis boundary condition information is written to the output database only when some nodal output variables are requested as field output.

Tracer Particle Output from Abaqus/Explicit

Products: Abaqus/Explicit

References:

- *About Output*
- **OUTPUT*

Overview

In Abaqus/Explicit tracer particles can be used to obtain output at specific material points that may not correspond to a fixed location in the mesh if adaptive meshing or an Eulerian mesh is used.

Tracer particles follow the material motion throughout an analysis regardless of the mesh motion, which makes them ideal for use with adaptive meshing (see *Defining ALE Adaptive Mesh Domains in Abaqus/Explicit*) and during an Eulerian analysis (see *Eulerian Analysis*). Both nodal and element output can be obtained at tracer particles.

Defining Tracer Particles

You define the initial location of each tracer particle to be coincident with a node, called the “parent node.” These parent nodes are grouped into a tracer set; you must assign a name to the tracer set when you define the tracer particles. In Eulerian analyses parent nodes grouped into the same tracer set as the connected elements must belong to the same Eulerian section. Tracer particle output is not supported when Eulerian mesh motion is used.

Input File Usage: **TRACER PARTICLE*, TRACER SET=*tracer_set_name*
list of parent nodes (either node numbers or node set labels)

Particle Birth Stages

Sets of tracer particles can be released from the current locations of the parent nodes at multiple times during a step. Each release of tracer particles is referred to as a “particle birth.” After particle birth the tracer particles follow the motion of the associated material regardless of the motion of the mesh. You can indicate the number of particle birth stages in a step, *n*. One particle birth will occur at the beginning of the step, and the rest of the stages will be evenly spaced throughout the step. If you do not specify a number of particle birth stages, a single particle birth will occur at the beginning of the step.

Input File Usage: **TRACER PARTICLE*, TRACER SET=*tracer_set_name*, PARTICLE BIRTH STAGES=*n*

Tracer Particles in the Output Database

Tracer sets will appear as both node and element sets in the output database. If a tracer set has multiple birth stages, additional node and element sets will be created that group all the tracer particles associated with a given birth stage. These subsets are named by appending the birth stage number to the tracer set name. For example, if a tracer set with the name `INLET` is defined with two particle birth stages, three node sets and three element sets will be created in the output database: `INLET Stage 1`, `INLET Stage 2`, and `INLET` (which contains all the nodes/elements from both `INLET Stage 1` and `INLET Stage 2`).

When tracer particles are used with adaptive meshing, internal field output requests are generated automatically for the requested output variables for all the elements or nodes in the domain that completely defines the space of possible tracer particle locations. This region is determined by Abaqus/Explicit and typically corresponds to the elements attached to the parent nodes and any intersecting adaptive mesh domains. The postprocessing calculator (see *The Postprocessing Calculator*) will compute the value of any requested output quantity at a tracer particle by interpolating the results from the element that encompasses the particle at the time of output.

When tracer particles are used in an Eulerian analysis, Abaqus/Explicit processes the output requests in the same way as for other node and element output; therefore, the postprocessing calculator is not used, and no additional internal requests are generated.

Requesting Output at Tracer Particles

You can request element or nodal output for a particular tracer set. Output will be given for all tracer particles that are associated with the specified tracer set name.

Input File Usage: Use one of the following options:

**NODE OUTPUT*, TRACER SET=*tracer_set_name*
**ELEMENT OUTPUT*, TRACER SET=*tracer_set_name*

Field Output at Tracer Particles

Displacement is the only valid field request for tracer particles. You can obtain the positions of the tracer particles in a specific tracer set by requesting displacements as nodal field output. Tracer particle displacements are output automatically if displacement output is requested for the entire model. You can use the node and element sets created for tracer particles in the output database to control the display of tracer particles in the Visualization module of Abaqus/CAE.

Input File Usage: Use both of the following options:

**OUTPUT*, FIELD
**NODE OUTPUT*, TRACER SET=*tracer_set_name*
U

History Output at Tracer Particles

Requesting history output for tracer particles is similar to requesting history output for elements and nodes. Any valid element integration point variable can be requested. U, V, A, and COORD are the only valid nodal requests. Whole element variables and element section variables cannot be requested. History data are available for a tracer particle only after its birth. When tracer particles are used in an Eulerian analysis, PRESS is the only valid element request.

When tracer particles are used with adaptive meshing, tracer particle history output request triggers an internal field output request for the desired variables for all the elements or nodes in the domain that completely defines the space of possible tracer particle locations.

Input File Usage: Use the following options:

**OUTPUT*, HISTORY
**NODE OUTPUT*, TRACER SET=*tracer_set_name*
**ELEMENT OUTPUT*, TRACER SET=*tracer_set_name*

Tracer Particle Propagation in Multiple Steps

Once defined, all tracer particles remain active in subsequent steps. However, no further particle births occur in the steps that follow the tracer set definition. You can define new tracer particles in subsequent steps by specifying a new tracer set name. The same tracer set name cannot be used more than once within an analysis.

Tracer Particle Deactivation

When used with adaptive meshing, individual tracer particles are deactivated if they flow out of the mesh across an Eulerian boundary or are currently tracking material points inside a failed element that has been deleted from the mesh. History data for tracer particles are zero at all times after deactivation.

When used in an Eulerian analysis, individual tracer particles are deactivated when they reach the Eulerian mesh boundary. They are also deactivated if the elements containing these tracer particles become void, which is usually caused by accumulated numerical error during interface reconstruction. Deactivated tracer particles have zero displacement.

Controlling the Output Frequency at Tracer Particles

The frequency of tracer particle output is controlled as described in *Controlling the Frequency of Output to the Output Database*.

Warning:

When tracer particles are used with adaptive meshing, requesting tracer set history output at a high frequency may cause the output database (`.odb`) to become large. The disk space required to store the field data is directly proportional to the size of the adaptive mesh domain and the number of tracer sets. The disk space usage is independent of the number of tracer particles in a tracer set. The output database file size is reduced after the postanalysis calculation is performed.

Integrated Output

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

In Abaqus/Explicit integrated output can be requested either over a surface or over an element set; in Abaqus/Standard integrated output can be requested over a surface.
An integrated output request is used to write the time history of variables such as the total force transmitted across a surface, the total mass of an element set, or the percentage change of the total mass of an element set.

Selecting the Integrated Output Variables

The integrated variables that can be written to the output database in Abaqus/Explicit are defined in *Integrated Variables*. The integrated variables that can be written to the output database in Abaqus/Standard are defined in *Section Variables*.

Input File Usage: **INTEGRATED OUTPUT*
list of output variables

Abaqus/CAE Usage: Step module: history output request editor: **Select from list below**

Selecting the Surface over Which Integrated Output Is Required

You can specify the surface directly for an integrated output request. Alternatively, you can associate an integrated output section that identifies the surface (see *Integrated Output Section Definition*) with the integrated output request.

Integrated output can be requested for a surface that includes facets, edges, or ends of various types of deformable elements. The surface can include facets of three-dimensional solid elements and continuum shell elements; edges of two-dimensional solid elements, membrane elements, conventional shell, and surface elements; and ends of beam elements, pipe elements, and truss elements.

Specifying the Surface for Integrated Output Directly

If you specify the surface for an integrated output request directly, any vector output variables are given with respect to a fixed global coordinate system and the total moment transmitted across the surface, SOM, is computed about the fixed global origin. See *Element-Based Surface Definition* for information on defining element-based surfaces.

Input File Usage: Use both of the following options:

**SURFACE*, NAME=*surface_name*, TYPE=ELEMENT
**INTEGRATED OUTPUT*, SURFACE=*surface_name*

Abaqus/CAE Usage: You cannot specify the surface for an integrated output request directly in Abaqus/CAE; you must create an integrated output section as described below.

Specifying the Surface through an Integrated Output Section Definition

If you associate an integrated output section definition with an integrated output request, the integrated output variables can be obtained in a local coordinate system that can translate and rotate with the deformation (see *Figure 1*). In addition, the total moment transmitted across the surface, SOM, can be computed about a moving location.

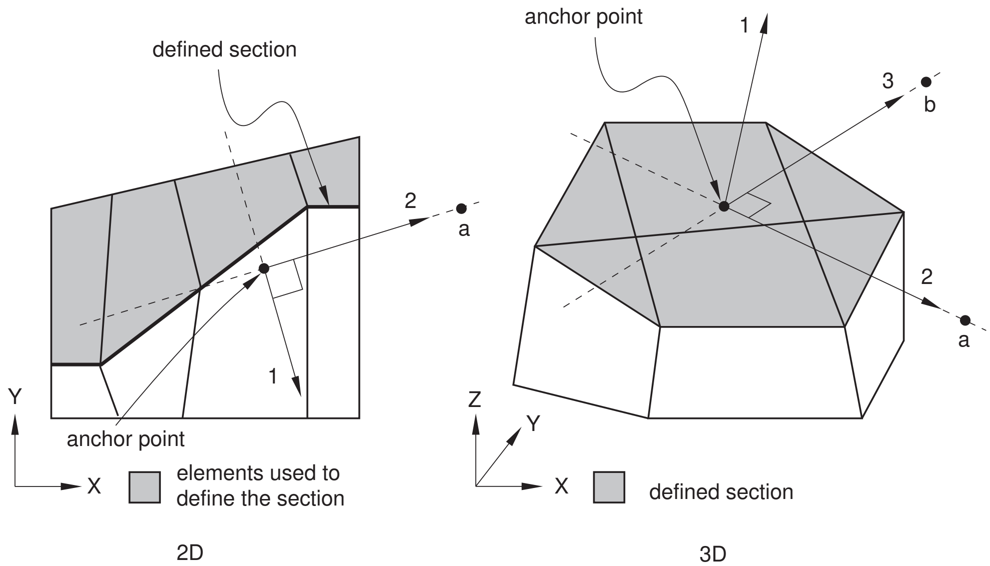


Figure 1: User-defined local coordinate system.

Input File Usage: Use both of the following options:

**INTEGRATED OUTPUT SECTION*, NAME=*section_name*, SURFACE=*surface_name*
**INTEGRATED OUTPUT*, SECTION=*section_name*

Abaqus/CAE Usage: Step module:

Output->Integrated Output Sections->Create: **Name:***section_name*: select regions for the surface

History output request editor: **Domain: Integrated output section:** *section_name*

Requesting Integrated Output for "Force-Flow" Studies

To study the "force-flow" through various paths in a model, you must create interior surfaces that cut through one or more regions (similar to a cross-section) so that you can request integrated output of the total force transmitted across these surfaces. You can create such interior surfaces over the element facets, edges, or ends by cutting through one or more regions of the model with a plane; see *Creating Interior Cross-Section Surfaces* for more information.

Input File Usage: Use both of the following options:

**SURFACE*, NAME=*surface_name*, TYPE=CUTTING SURFACE
**INTEGRATED OUTPUT*, SURFACE=*surface_name*

Abaqus/CAE Usage: You cannot specify the surface for an integrated output request directly in Abaqus/CAE; you must create an integrated output section as described above.

Requesting Integrated Output over an Element Set in Abaqus/Explicit

You can request integrated output over an element set to output its total mass, the percentage change of its total mass, its average rigid body motion, or any combination of these variables. The element set must have been defined previously, and it can include any type of elements. Only dedicated integrated output quantities are supported for Eulerian or discrete particle element sets. These output quantities are defined in *Integrated Variables*.

Input File Usage: Use the following option to request integrated output over an element set:

**INTEGRATED OUTPUT*, ELSET=*element set name*

Abaqus/CAE Usage: Requesting integrated output over an element set is not supported in Abaqus/CAE.

Controlling the Output Frequency

The frequency of integrated output is controlled as described in *Controlling the Frequency of Output to the Output Database*.

Requesting Preselected Integrated Output

Preselected output variables are available only when the integrated output is requested over a surface. If integrated output is requested over an element set, you must specify the variables on the data line.

If the integrated output is requested over a surface, you can request the preselected integrated output variables SOF and SOM. In this case you can also specify additional variables as part of the output request. Alternatively, you can request all integrated variables applicable to the current procedure type. In this case any additional variables that you specify are ignored. If you do not request the preselected variables or all variables, you must specify the variables individually.

Input File Usage: Use the following option to request the preselected integrated output variables:

**INTEGRATED OUTPUT*, VARIABLE=PRESELECT
optional additional variables

Use the following option to request all integrated output variables relevant to the current procedure type:

**INTEGRATED OUTPUT*, VARIABLE=ALL

Use the following option to specify individual integrated output variables:

**INTEGRATED OUTPUT*
individual variables

Abaqus/CAE Usage: Step module: history output request editor: **Preselected defaults** or **All**

Limitations When Using Integrated Output Requests

Integrated output requests over a surface are subject to the following limitations:

- Integrated output can be requested over a surface that includes facets, edges, or ends of various types of deformable elements. The surface can include facets of three-dimensional solid elements and continuum shell elements; edges of two-dimensional solid elements, membrane elements, conventional shell, and surface elements; and ends of beam elements, pipe elements, and truss elements. The surface should not contain facets of axisymmetric elements or facets of rigid elements.
- When defining the surface, elements on only one side of the surface must be used. Abaqus/Explicit computes the integrated output variables using the stresses and hourglass-mode forces in elements underlying the surface as in a free-body diagram.
- The defined surface must cut completely through the mesh, form a closed surface, or be on the exterior of the body. *Figure 2* presents some typical cases of valid surfaces. If the surface cuts only partially through the mesh, a valid free-body diagram cannot be isolated (see *Figure 3*) and incorrect answers may be computed.

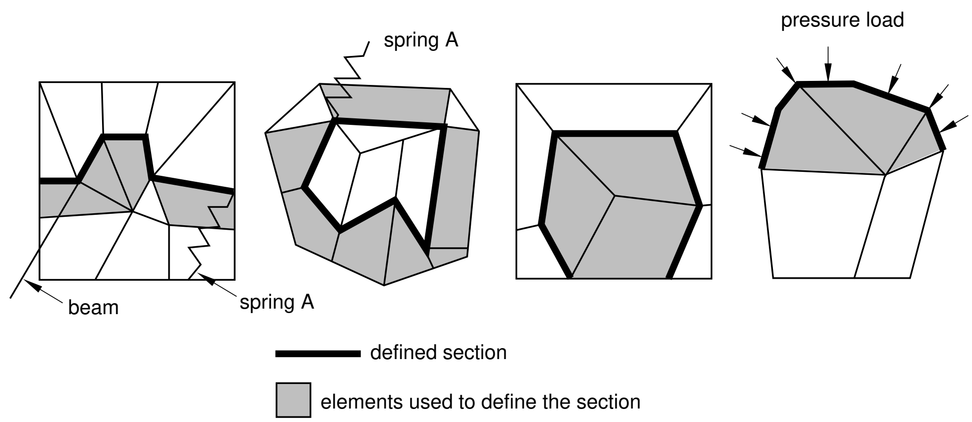


Figure 2: Valid section definitions.

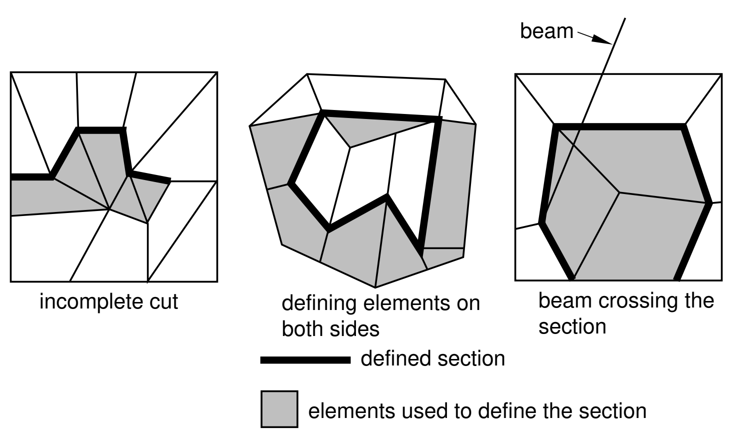


Figure 3: Invalid section definitions.

- Elements attached to the surface can be on either side of the surface but must not cross the defined surface. *Figure 3* presents a few invalid cases.
- The total force and the total moment in the section are computed based only on the stresses (internal forces) in the identified elements. Thus, inaccurate results may be obtained if distributed body loads are present in these elements since their effect on the total force in the section is not included. Common examples are the inertial loading in dynamic analyses, gravity loads, distributed body forces, and centrifugal loads. In these cases the total force in the section may depend on the choice of elements used to define the section as illustrated in *Figure 4*(a).

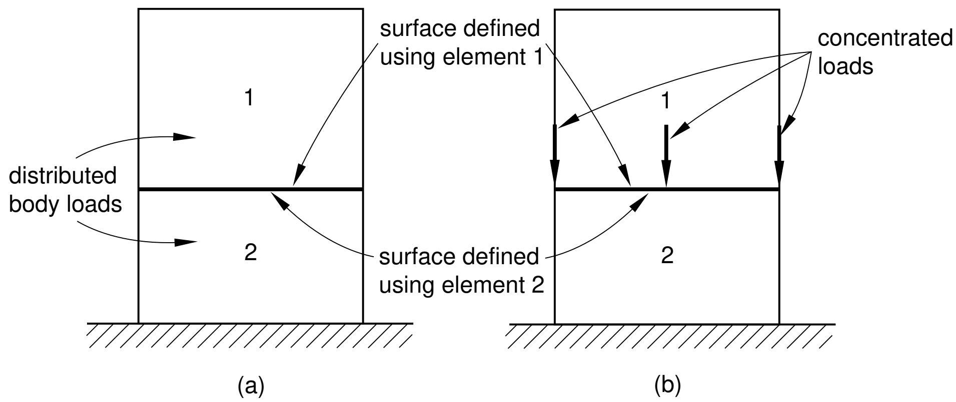


Figure 4: Total force in the section.

Assuming that gravity loading is the only active load, the element stresses will be different in the two elements. Hence, if the same surface is defined first using element 1 and then using element 2, different answers for the total force will be obtained. In a similar way the effects of any distributed body fluxes (heat, electrical, etc.) prescribed in the identified elements are not included.

- Depending on which side of the surface is used to define the section, different answers will be obtained in analyses similar to the case illustrated in *Figure 4*(b). Assuming a quasi-static analysis with the concentrated loads shown in the figure being the only active loads, a zero total force is reported if the surface is defined using element 1 and a nonzero force equal to the sum of the concentrated loads is obtained if the surface is defined using element 2.
- If the nodes that are part of the integrated output surface also participate in constraints (such as a tie constraint), the constraint force or flux is not included in the integrated output.

Total Energy Output

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

You can output the total energy of the model or of a specific element set to the output database.

Energy output is available only as history output. Energy output requests are not available for the following procedures:

- Eigenvalue buckling prediction
- Complex eigenvalue extraction

Additional information is available for the following procedures:

- Frequency extraction: *Energy Output*
- Mode-based steady-state analysis: *Total Energy Output*

Selecting the Energy Output Variables

The energy variables that can be written to the output database are defined for Abaqus/Standard in *Total Energy Output Quantities* and for Abaqus/Explicit in *Total Energy Output*.

Input File Usage: **ENERGY OUTPUT*
list of output variables

Abaqus/CAE Usage: Step module: history output request editor: **Select from list below**

Selecting the Element Set for Which Total Energy Output Is Required

You can specify the element set for which total energy output is being requested. In this case the energies are summed for all the elements in the specified set. You cannot specify an element set for the following procedures:

- Transient modal dynamic analysis
- Response spectrum analysis
- Random response analysis

The following energies are not available as element set quantities: ALLCCDW, ALLCCE, ALLCCEN, ALLCCET, ALLCCSD, ALLCCSDN, ALLCCSDT, ALLFC, ALLFD, ALLKL, ALLQB, ALLWK, ALLVDM, ALLHDM, and ETOTAL.

If you do not specify an element set, the total energies for the whole model will be output. If total energy output for both the whole model and for different element sets is desired, the energy output requests must be repeated: once without a specified element set to request energy output for the whole model and once for each specified element set.

Input File Usage: **ENERGY OUTPUT*, ELSET=*element_set_name*

Abaqus/CAE Usage: Step module: history output request editor: **Domain: Set:** *set_name*

Controlling the Output Frequency

The frequency of energy output is controlled as described in *Controlling the Frequency of Output to the Output Database*.

Requesting Preselected Output

You can request the preselected, procedure-specific energy output variables described in *Preselected Output Requests*. In this case you can specify additional variables as part of the output request.

Alternatively, you can request all energy variables applicable to the current procedure and material type. In this case any additional variables you specify are ignored.

Input File Usage: Use the following option to request the preselected energy output variables:

**ENERGY OUTPUT*, VARIABLE=PRESELECT

Use the following option to request all applicable energy output variables:

**ENERGY OUTPUT*, VARIABLE=ALL

Abaqus/CAE Usage: Step module: history output request editor: **Preselected defaults** or **All**

Defining Sensors

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

For nodal, connector element, and some whole surface contact output variables, history output requests can be used to define sensors.
Sensors are named entities that are intended to be used to model physical sensors such as the total force or displacement of a hydraulic piston, the motion of a given point on a structure, or the acceleration as measured by an accelerometer. Sensor values can be fed back into the model to produce actuation that is a function of the sensed quantity thus allowing for modeling of control engineering aspects of your system.

You can use sensors in user subroutine *UAMP* or *VUAMP* to define a customized amplitude that is a function of sensor values at the end of the previous increment as shown in *VUAMP* and illustrated in the example in *Crank mechanism*. Alternatively, you can export sensor values to a Functional Mockup Unit (FMU) and import computed actuation information; that is, the current amplitude value of an amplitude function in a co-simulation (see *System-Level Modeling between Logical and Physical Interactions*). In these cases you can use the amplitude function to actuate any Abaqus feature that can reference an amplitude, such as concentrated loads, boundary conditions, connector motion/load, distributed pressure, and material properties via field variables.

A sensor must be uniquely associated with a particular scalar output variable (U1, CTF3, etc.) and can be defined using history output requests by following some simple rules. The sensor name is specified in the history output definition, and one and only one nodal output, element output, or whole surface request can be defined for each sensor. For whole surface contact or contact pair output requests, only the magnitude and the center of the total force due to contact pressure (CFNM and XN, respectively) are supported. Because the named sensor must be a unique real number at a given time, the node set or element set used to define the sensor must contain only one member. Moreover, regardless of the user-specified output frequency, sensors are computed at every increment during the analysis. However, they are written to the output database according to the user-specified frequency.

Input File Usage: Use the following options to define a sensor using element output:

**OUTPUT*, HISTORY, SENSOR, NAME=*name*
**ELEMENT OUTPUT*
element output variable

Use the following options to define a sensor using nodal output:

**OUTPUT*, HISTORY, SENSOR, NAME=*name*
**NODE OUTPUT*
nodal output variable

Use the following options to define a sensor using contact output:

**OUTPUT*, HISTORY, SENSOR, NAME=*name*
**CONTACT OUTPUT*
contact or contact pair output variable (CFNM or XN)

Abaqus/CAE Usage: Step module: history output request editor: **Domain: Set:** *name*, toggle on **Include sensor when available**

Filtering Output and Operating on Output in Abaqus/Explicit

Products: Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

You can pre-filter element and nodal field output and element, nodal, contact, integrated, and fastener interaction history output before it is written to the output database.
You can also operate on filtered or unfiltered (raw) output data to extract the maximum, minimum, or absolute maximum of the output variables over time. In addition, you can set a limit value for the output variables, and you can stop the analysis at the time this limit is reached. For field output the time at which the maximum, minimum, and absolute maximum were reached or the time when the limit was reached is output by default for each output variable.

If you filter a field output request that includes many output variables and applies to the entire model, the memory requirements and the running time will both increase. For common output requests consisting of a few element output variables and a few nodal output variables the memory requirements and the running time will not increase substantially.

Defining a Low-Pass Infinite Impulse Response Digital Filter

You can define three types of low-pass Infinite Impulse Response filters as part of the model definition. Typical magnitude curves for analog type filters are presented in *Figure 1*, where Ω_c represents the normalized cutoff frequency, which is the ratio of the cutoff frequency to the sampling frequency (the sampling frequency is the inverse of the time increment).

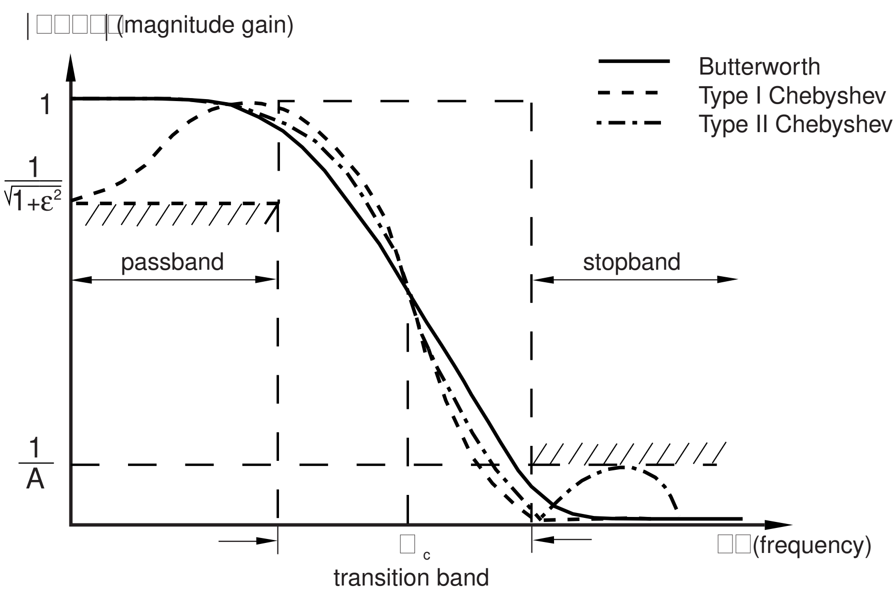


Figure 1: Typical magnitude curves for low-pass filters.

The Butterworth filter is very common; its response in the pass band is known as maximally flat. The Type I Chebyshev filter has a sharper transition between the pass band and the stop band, but it has a ripple in the pass band. The Type II Chebyshev filter also has a sharper transition between the pass band and the stop band than a Butterworth filter of the same order, but it has a ripple in the stop band. The higher the order of the filter, the narrower the transition band. However, the computational cost increases as the order increases. In addition, for high-order filters the phase lag, which is the time delay between the filtered and unfiltered signal, may become significant. For most applications filter orders of two or four are sufficiently accurate.

To define a Butterworth filter, you must specify the cutoff frequency, f_c , and the filter order, *N*. Since the implementation of the filters is done using cascades of second-order sections, Abaqus expects an even number for the filter order. If you specify an odd number for the order, the order will be increased internally to the next even number. The default value for the order is two, and the highest order that can be prescribed is twenty. For the Chebyshev filters you must also specify an additional parameter, the ripple factor. The ripple factor is equal to ϵ for a Type I Chebyshev filter and is equal to $1/A$ for a Type II Chebyshev filter (see *Figure 1*).

No checks are performed to ensure that the cutoff frequency is appropriate; for example, Abaqus does not check that only the noise of the signal is eliminated. You need to know the range of the physical frequencies that are expected in the solution, and you must prescribe a cutoff frequency greater than these frequencies. In addition, the cutoff frequency should be less than half the sampling frequency; otherwise, no filtering is performed. Abaqus internally remaps (using a quadratic interpolation) the output raw data so that the filtering can satisfy the constant time-increment (sampling) requirement.

Defining the Filter Type

You must assign each filter definition a name that can be used to refer to the filter from an output request.

Input File Usage: Use one of the following options to define a filter:

**FILTER*, NAME=*filter_name*, TYPE=BUTTERWORTH
**FILTER*, NAME=*filter_name*, TYPE=CHEBYS1
**FILTER*, NAME=*filter_name*, TYPE=CHEBYS2

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Name:** *filter_name*; **Butterworth**, **Type I Chebyshev**, or **Type II Chebyshev**

Start-Up Conditions for the Filter

By default, the values of the variables at time zero (zero increment) are used as the initial conditions (or start-up conditions); however, you can change this initial value.

Input File Usage: Use the following option to use the default initial conditions:

**FILTER*, NAME=*filter_name*, TYPE=*filter_type*, START CONDITION=DC

Use the following option to specify the initial variable values:

**FILTER*, NAME=*filter_name*, TYPE=*filter_type*,
START CONDITION=USER DEFINED

Abaqus/CAE Usage: You cannot specify the initial variable values in Abaqus/CAE.

Filtering Using the Low-Pass Infinite Impulse Response Filters

To pre-filter element, nodal, contact, or integrated history output or element and nodal field output based on one of the low-pass Infinite Impulse Response filters that you defined, you refer to this filter by name from the output request.

Input File Usage: Use the following option to apply a filter to an output request:

**OUTPUT*, FILTER=*filter_name*

Abaqus/CAE Usage: Step module: field or history output request editor: **Apply filter:** *filter_name*

Filtering the Output Based on the Time Interval

For history output you can request that Abaqus/Explicit create an antialiasing filter that is internally based on the time interval specified in the output request. The cutoff frequency is set internally to one-sixth of the time frequency (the time frequency is the inverse of the time interval, t, used for history output). If no time intervals are specified, the default number of history output intervals is used to create the cutoff frequency of the filter. You can also use antialiasing filters for a field output request, but in this case the cutoff frequency is set to one-sixth of a time frequency corresponding to two hundred time intervals per step if less than two hundred field frames are requested. If more than two hundred field frames are requested, the cutoff frequency is set to one-sixth of the requested time frequency. The antialiasing filter is a second-order Butterworth type and a filter definition is not required.

Abaqus/Explicit does not check whether the specified time interval for history output provides an appropriate cutoff frequency to build the internal filter. You should know approximately how many data points are required to describe your history curve (or signal) accurately, and Abaqus/Explicit will give you the most physical (un-aliased) representation of the signal for that number of points. Similarly for field output Abaqus/Explicit does not check whether the cutoff corresponding to two hundred sampling intervals or more (if you request more than two hundred frames) is appropriate for your analysis. If a lower (or higher) cutoff frequency is needed, you should define the filter in the model data.

Filtering Field Output or History Output Written at Time Intervals

You can apply a filter to a field output request or a history output request written at intervals of time in your analysis.

Input File Usage: Use one of the following options:

**OUTPUT*, FIELD, FILTER=ANTIALIASING, TIME INTERVAL=*t*

**OUTPUT*, HISTORY, FILTER=ANTIALIASING, TIME INTERVAL=*t*

Abaqus/CAE Usage: Step module: field or history output request editor: **Frequency**: **Every *x* units of time**: *t*, **Apply filter**: **Antialiasing**

Filtering Field Output Written at Evenly Spaced Intervals of Time

You can apply a filter to a field output request written at evenly spaced time intervals in your analysis.

Input File Usage: **OUTPUT*, FIELD, FILTER=ANTIALIASING, NUMBER INTERVAL=*n*

Abaqus/CAE Usage: Step module: field output request editor: **Frequency**: **Evenly spaced time intervals**, **Interval**: *n*, **Apply filter**: **Antialiasing**

Requesting Maximum, Minimum, or Absolute Maximum Values for an Output Request

You can apply a filter to a field output request or a history output request to obtain the maximum, minimum, or absolute maximum values for each variable in the output request. The absolute maximum option enables you to obtain the largest absolute value, negative or positive, for each variable in the output request. Abaqus evaluates maximum, minimum, or absolute maximum values at every increment during the analysis and reports these values at the time given by the output interval specified in the output request. For field output requests the last output frame will contain the maximum (or absolute maximum) value and minimum value over the entire step; the intermediate frames will show the maximum, minimum, or absolute maximum value up to the frame time. An additional output variable containing the time when the maximum, minimum, or absolute maximum occurred is output automatically for each output variable requested. This time output is written by default (and it cannot be suppressed).

For field output requests Abaqus filters by default each component of tensor and vector quantities of output variable independently and provides separate maximum, minimum, or absolute maximum values for each component of the variable. You can, however, request the maximum or minimum value or apply a limit value to an invariant such as Mises stress for element output or magnitude for nodal output (see *Applying Bounding Values to Invariants*).

Requesting Maximum, Minimum, or Absolute Maximum Values for Filtered Output

You can define a low-pass digital filter that returns the maximum, minimum, or absolute maximum value for output requests to which it is applied.

Input File Usage: Use one of the following options:

**FILTER*, TYPE=*filter_type*, OPERATOR=MAX
**FILTER*, TYPE=*filter_type*, OPERATOR=MIN
**FILTER*, TYPE=*filter_type*, OPERATOR=ABSMAX

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Butterworth**, **Type I Chebyshev**, or **Type II Chebyshev**: **Determine bounding value**: **Maximum**, **Minimum**, or **Absolute maximum**

Requesting Maximum, Minimum, or Absolute Maximum Values for Unfiltered Output

You can define a filter that returns the maximum, minimum, or absolute maximum value for output requests to which it is applied without performing any digital filtering of the data.

Input File Usage: Use one of the following options:

**FILTER*, OPERATOR=MAX
**FILTER*, OPERATOR=MIN
**FILTER*, OPERATOR=ABSMAX

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Type**: **Operator**: **Determine bounding value:Maximum**, **Minimum**, or **Absolute maximum**

Setting an Upper or Lower Limit on Variables in an Output Request

You can apply a filter to a field output request or a history output request to prescribe a bounding value for the variables in the output request. If any of the variables in the output request reach a value higher than the maximum limit, lower than the minimum limit, or greater than the absolute maximum limit, Abaqus returns the limiting value. The time at which the limit was reached is output separately for each requested variable. This time output is written by default (and it cannot be suppressed). You can also request an additional field output frame when the limiting value is reached.

Setting an Upper Limit or a Lower Limit for Filtered Output

You can define a low-pass digital filter that enforces an upper or lower bound for the variables in the output requests to which it is applied.

Input File Usage: **FILTER*, TYPE=*filter_type*, OPERATOR=*operator_type*, LIMIT=*value*

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Type**: **Butterworth**, **Type I Chebyshev**, or **Type II Chebyshev**: **Determine bounding value:Maximum**, **Minimum**, or **Absolute maximum**: toggle on **Bounding value limit:***value*

Setting an Upper Limit or a Lower Limit for Unfiltered Output

You can define a filter that enforces an upper or lower bound for the variables in the output requests to which it is applied but that does not perform any Butterworth or Chebyshev filtering of the data.

Input File Usage: **FILTER*, OPERATOR=*operator_type*, LIMIT=*value*

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Type**: **Operator**: **Determine bounding value:Maximum**, **Minimum**, or **Absolute maximum**: toggle on **Bounding value limit:***value*

Requesting an Extra Output Frame for Filtered or Unfiltered Output

You can capture, on demand, the analysis state through field output requests when the value of an output variable reaches a specified upper or lower bound. All of the on demand field output is written soon (usually within two increments) after the increment in which the bounding value is reached. Once the criterion is met, the feature is turned off automatically to avoid repeated on demand output. If necessary, you can use the feature multiple times with varying bounding values.

Input File Usage: **FILTER*, OPERATOR=*operator_type*, LIMIT=*value*, EXTRA OUTPUT FRAME=YES or NO

Abaqus/CAE Usage: Requesting an extra output frame for filtered or unfiltered output is not supported in Abaqus/CAE.

Stopping an Analysis or Concluding a Step When an Output Variable Reaches a Prescribed Limit

You can apply a filter to a field output request or a history output request that stops the analysis or concludes a step when the value of any variable in the output request reaches a specified upper bound or lower bound.

Stopping an Analysis or Concluding a Step of Filtered Output When a Variable Reaches a Prescribed Limit

You can define a low-pass digital filter that stops the analysis or concludes a step if any of the variables in the output requests to which it is applied reach a prescribed limit.

Input File Usage: **FILTER*, TYPE=*filter_type*, OPERATOR=*operator_type*, LIMIT=*value*, HALT=ANALYSIS or STEP

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Butterworth**, **Type I Chebyshev**, or **Type II Chebyshev**: **Determine bounding value:Maximum**, **Minimum**, or **Absolute maximum**: toggle on **Bounding value limit**: *value*: toggle on **Stop analysis upon reaching limit**

Concluding the current step and continuing with the next step is not supported.

Stopping an Analysis or Concluding a Step of Unfiltered Output When a Variable Reaches a Prescribed Limit

You can define a filter that does not perform any Butterworth or Chebyshev filtering of your output data and stops the analysis or concludes a step if any of the variables in the output requests to which it is applied reach a prescribed limit.

Input File Usage: **FILTER*, OPERATOR=*operator_type*, LIMIT=*value*, HALT=ANALYSIS or STEP

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Type**: **Operator**: **Determine bounding value:Maximum**, **Minimum**, or **Absolute maximum**: toggle on **Bounding value limit**: *value*: toggle on **Stop analysis upon reaching limit**

Concluding the current step and continuing with the next step is not supported.

Applying Bounding Values to Invariants

By default, each component of a tensor or vector quantity is filtered individually and the maximum, minimum, or absolute maximum value and the limiting values are reported separately for each component. You can, however, apply a filter directly to an invariant. In this case Abaqus internally monitors the invariant you specified. Abaqus still writes the components to the output database, but these components correspond to the maximum, minimum, or limiting values of the invariant. *Table 1* shows which invariants are available for output variable categories.

Table 1: Invariants available for output variable categories.

Category	First invariant	Second invariant	MaxP	IntermP	MinP
All nodal vector output	Magnitude	–	—	—	—
Stress element output	Mises	Press	SP3	SP2	SP1

Category	First invariant	Second invariant	MaxP	IntermP	MinP
Logarithmic strain output			LEP3	LEP2	LEP1
Nominal strain output			NEP3	NEP2	NEP1
Thermal strain output			THE3	THE2	THE1

Applying Bounding Values to Invariants of Filtered Output

You can define a low-pass digital filter that filters the invariant.

Input File Usage: **FILTER*, TYPE=*filter_type*, OPERATOR=*operator_type*, LIMIT=*value*, INVARIANT=FIRST, SECOND, MAXP, INTERMP, or MINP

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Type**: **Butterworth**, **Type I Chebyshev**, or **Type II Chebyshev**; toggle on **Bounding value limit**: *value*: **Invariant**: **First** or **Second**

You cannot request maximum, intermediate, and minimum principal stresses for invariants in Abaqus/CAE.

Applying Bounding Values to Invariants of Unfiltered Output

You can define a filter that does not perform any Butterworth or Chebyshev filtering of your output data and filters the invariant.

Input File Usage: **FILTER*, OPERATOR=*operator_type*, LIMIT=*value*, INVARIANT= FIRST or SECOND

Abaqus/CAE Usage: Step module: **Tools->Filter->Create**: **Type**: **Operator**; toggle on **Bounding value limit**: *value*: **Invariant**: **First** or **Second**

Output Variables Available for Filtering

Low-pass Infinite Impulse Response filters such as Butterworth and Chebyshev filters are intended for filtering of output variables susceptible to noise, such as accelerations and reaction forces or, to a lesser degree, stress and strain. However, digital filtering is allowed for most element and nodal output variables, and you can apply bounding values on unfiltered data for nearly all element and nodal output variables. *Table 2* shows the set of output variables that cannot be digitally filtered but to which you can apply bounding values, and *Table 3* shows the set of output variables for which neither digital filtering nor application of bounding values are allowed.

Table 2: Output variables to which bounding values can be applied but digital filtering cannot be applied.

Category	Output variables
Tensors and invariants	PEEQ
State and field variables	TEMP, FV
Energy densities	ENER, SENER, PENER, CENER, VENER, DMENER
Additional plasticity quantities	PEQC
Cracking model quantities	CKSTAT

Category	Output variables
Whole element variables	EDT, EMSF, ELEDEN, ESEDEN, EPDDEN, ECDDEN, EVDDEN, EASEDEN, EIHEDEN, EDMDDEN, ELEN, ELSE, ELCD, ELPD, ELVD, ELASE, ELIHE, ELDMD, ELDC, STATUS
Nodal output variables	NT, COORD

Table 3: Output variables that cannot be digitally filtered or modified with bounding values.

Category	Output variables
Cracking model quantities	CRACK
Element face variables	STAGP, TRNOR, TRSHR
Whole element variables	GRAV, BF, SBF, P
Nodal output variables	CF
Total energy output	ALLAE, ALLCD, ALLFD, ALLIE, ALLKE, ALLPD, ALLSE, ALLVD, ALLWK, ALLIHE, ALLHF, ALLDMD, ALLDC, ALLFC, ALLPW, ALLCW, ALLMW, ETOTAL
Time increment and mass output	DT, DMASS, CDMASS, SSPEEQ, SSSPRD, SSFORC, SSTORQ

Modal Output from Abaqus/Standard

Products: Abaqus/Standard Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

You can output generalized coordinate (modal amplitude and phase values during modal dynamic procedures to the output database.
(See *About Dynamic Analysis Procedures* for an overview of the modal dynamic procedures available in Abaqus/Standard.) Modal output is available only as history output.

Controlling the Frequency of Output

The frequency of modal output is controlled as described in *Controlling the Output Frequency in Abaqus/Standard*.

Requesting Output

You can choose to request all modal variables applicable to the current procedure and material type. In this case any additional variables you specify are ignored.

Input File Usage: **MODAL OUTPUT*, VARIABLE=ALL

Abaqus/CAE Usage: Step module: history output request editor: **All**

Writing Surface Output to the Output Database

Products: Abaqus/Standard Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

You can write variables associated with surfaces in contact, coupled thermal-electrical-structural (Abaqus/Standard only), coupled temperature-displacement (Abaqus/Standard only), coupled thermal-electrical, and crack propagation problems to the output database.
Multiple output requests can be used to customize requests among interactions, surfaces, or node sets.

For surface variables associated with cavity radiation, see *Cavity Radiation Output in Abaqus/Standard*.

Use element output requests (see *Writing Element Output to the Output Database*) to obtain database output for contact elements (such as gap elements; see *Gap Contact Elements*).

In Abaqus/Standard contact history output cannot be saved in a linear perturbation step with frequency extraction.

Displacement nodal output is generated automatically in Abaqus/Explicit when requesting surface output.

Selecting the Surface Output Variables

The surface variables that can be written to the output database are listed in *Surface Variables* (Abaqus/Standard) and *Surface Variables* (Abaqus/Explicit).

Input File Usage: **CONTACT OUTPUT*
list of output variables

Abaqus/CAE Usage: Step module: field or history output request editor: **Select from list below**

Limiting the Extent of a Surface Output Request in Abaqus/Standard

Output requests apply to general contact and all contact pair interactions in a model by default in Abaqus/Standard. Options to limit an output request to certain interactions are discussed below.

Limiting Output to a Node Set in Abaqus/Standard

You can limit a surface output request to apply to a subset of surface nodes involved in contact pairs or general contact in Abaqus/Standard.

Input File Usage: **CONTACT OUTPUT*, NSET=*node_set_name*

Abaqus/CAE Usage: Step module: field or history output request editor: **Domain: Interaction:** *contact_interaction_name*

Limiting Output for Contact Pairs Based on Secondary and Main Surface Names in Abaqus/Standard

You can limit output to certain contact pairs based on surface names. If you specify both the secondary and main surface names, the output request is limited to a specific contact pair. If you specify the secondary surface but not the main surface, output is written for all contact pairs that involve the specified secondary surface. If you also specify a node set, the applicability of an output request is further limited (that is, the output request will generate output only for certain nodes of a certain contact pair (or pairs). Output requests with a specific secondary and/or main surface role specified will not generate output for general contact.

Input File Usage: **CONTACT OUTPUT*, MAIN=*main*, SECONDARY=*secondary*, NSET=*node_set_name*

Abaqus/CAE Usage: Step module: field or history output request editor: **Domain: Interaction:** *contact_interaction_name*

Limiting Output for Cracked Surfaces in Enriched Elements Based on Surface Name in Abaqus/Standard

You can limit output requests to certain cracked surfaces in enriched elements based on surface names.

Input File Usage: **CONTACT OUTPUT*, SURFACE=*surface_name*

Abaqus/CAE Usage: You cannot limit surface field output for cracked surfaces in enriched elements in Abaqus/CAE.

Limiting the Extent of a Surface Field Output Request in Abaqus/Explicit

Field output requests apply to general contact and all contact pair interactions in a model by default in Abaqus/Explicit. Options to limit a surface field output request to certain interactions are discussed below.

Limiting Surface Field Output to a Contact Pair Set in Abaqus/Explicit

In Abaqus/Explicit you can select the contact pairs for which surface field output is desired. Surface output is contact pair-specific, so that contact output for a particular surface involved in a selected contact pair will include only the contributions from that contact pair if the surface is involved in multiple contact pairs. Surface output is available only for discrete (node-based or element-based) surfaces; it is not available for any analytical surfaces within a contact pair.

Input File Usage: Use the following option to request surface field output for a particular contact pair set:

**CONTACT OUTPUT*, CPSET=*contact_pair_set_name*

Abaqus/CAE Usage: Step module: field output request editor: **Domain: Interaction:** *contact_interaction_name*

Limiting Surface Field Output to General Contact in Abaqus/Explicit

You can limit surface field output requests to apply only to general contact in Abaqus/Explicit, but you cannot further limit this output to a subset of the general contact domain.

Input File Usage: **CONTACT OUTPUT*, GENERAL CONTACT

Abaqus/CAE Usage: You cannot limit surface field output to general contact in Abaqus/CAE.

Limiting Surface Field Output to a Single Surface in Abaqus/Explicit

You can limit surface field output requests to a single surface in the general contact domain in Abaqus/Explicit. The contact output for the specified surface will include all the contributions from other contact surfaces interacting with the surface.

Input File Usage: **CONTACT OUTPUT*, SURFACE=*surface_name*

Abaqus/CAE Usage: You cannot limit a single surface output to general contact in Abaqus/CAE.

Limiting Surface Field Output to Pairwise Surfaces in Abaqus/Explicit

You can specify a pair of surfaces in the general contact domain in Abaqus/Explicit for which the interactions on one surface due to the contact with another surface will be output. This type of output cannot be used for surfaces involving Eulerian regions. The following contact output variables are not supported with this type of output: CDISP, CTANDIR, CSLIPR, CORIENT, CFRICWORK, and CSTATUS.

Input File Usage: **CONTACT OUTPUT*, SURFACE=*first_surface_name*, SECOND SURFACE=*second_surface_name*

Abaqus/CAE Usage: You cannot limit pairwise surface output to general contact in Abaqus/CAE.

Specifying Surface History Output Regions in Abaqus/Explicit

You must specify an interaction to which a surface history output request applies with one of the methods discussed below.

Specifying Surface History Output by Contact Pair Set in Abaqus/Explicit

In Abaqus/Explicit you can select the contact pairs for which surface history output is desired. Surface output is contact pair-specific, so that contact output for a particular surface involved in a selected contact pair will include only the contributions from that contact pair if the surface is involved in multiple contact pairs. Surface output is available only for discrete (node-based or element-based) surfaces; it is not available for any analytical surfaces within a contact pair.

Input File Usage: Use the following option to request surface history output for a particular contact pair:

**CONTACT OUTPUT*, CPSET=*contact_pair_set_name*

Abaqus/CAE Usage: Step module: history output request editor: **Domain: Interaction:** *contact_interaction_name*

Specifying Whole Surface History Output in Abaqus/Explicit

You can specify a surface in the general contact domain for which whole surface contact force resultants will be output. Whole surface contact force resultants for a surface in the general contact domain are available only as history output.

Input File Usage: **CONTACT OUTPUT*, SURFACE=*surface_name*

Abaqus/CAE Usage: Step module: history output request editor: **Domain: General contact surface:** *surface_name*

Specifying Pairwise Surface History Output in Abaqus/Explicit

You can specify a pair of surfaces in the general contact domain for which the resultant contact forces on one surface due to the contact with another surface will be output. The contact force resultants in this case consider only the contact interactions between the two specified surfaces. This type of output cannot be requested for surfaces involving Eulerian regions.

Input File Usage: **CONTACT OUTPUT*, SURFACE=*first_surface_name*, SECOND SURFACE=*second_surface_name*

Abaqus/CAE Usage: You cannot request surface history output for a pair of surfaces in Abaqus/CAE.

Specifying Surface History Output by Fastened Node Set in Abaqus/Explicit

You can select a fastened node set for which bond history output is desired:

Input File Usage: Use the following option to request surface history output for a particular fastened node set:

**CONTACT OUTPUT*, NSET=*node_set_name*

Abaqus/CAE Usage: You cannot request surface history output for a particular fastened node set in Abaqus/CAE.

Controlling the Output Frequency

The frequency of surface output is controlled as described in *Controlling the Frequency of Output to the Output Database*.

Requesting Preselected Output

You can request the preselected, procedure-specific surface output variables described in *Preselected Output Requests*. In this case you can specify additional variables as part of the output request.

Alternatively, you can request all surface variables applicable to the current procedure. In this case any additional variables you specify are ignored.

Input File Usage: Use the following option to request the preselected surface output variables:

**CONTACT OUTPUT*, VARIABLE=PRESELECT

Use the following option to request all applicable surface output variables:

**CONTACT OUTPUT*, VARIABLE=ALL

Abaqus/CAE Usage: Step module: field or history output request editor: **Preselected defaults** or **All**

Time Incrementation Output in Abaqus/Explicit

Products: Abaqus/Explicit Abaqus/CAE

References:

- *About Output*
- **OUTPUT*
- *Understanding output requests*

Overview

You can output incrementation variables for an Abaqus/Explicit analysis to the output database. Incrementation output is available only as history output.

Selecting the Incrementation Output Variables

The following incrementation output variables are available in Abaqus/Explicit:

- Time increment size, DT
- Percent change in mass of the model due to mass scaling, DMASS
- Percent change in mass of the model due to only contact mass scaling, CDMASS
- Steady-state detection variables: SSPEEQ, SSSPRD, SSFORC, and SSTORQ

Input File Usage: **INCREMENTATION OUTPUT*
list of output variables

Abaqus/CAE Usage: Step module: history output request editor: **Select from list below**

Controlling the Output Frequency

The frequency of incrementation output is controlled as described in *Controlling the Frequency of Output to the Output Database*.

Requesting Preselected Output

You can request the preselected, procedure-specific incrementation output variables. In this case you can specify additional variables as part of the output request.

Alternatively, you can request all incrementation variables applicable to the current procedure type. In this case any additional variables you specify are ignored.

Input File Usage: Use the following option to request the preselected incrementation output variables:

**INCREMENTATION OUTPUT*, VARIABLE=PRESELECT

Use the following option to request all applicable incrementation output variables:

**INCREMENTATION OUTPUT*, VARIABLE=ALL

Abaqus/CAE Usage: Step module: history output request editor: **Preselected defaults** or **All**

Cavity Radiation Output in Abaqus/Standard

Products: Abaqus/Standard

References:

- *About Output*
- **OUTPUT*

Overview

You can request that cavity-, element-, or surface-based output such as radiation fluxes, view factor totals for a facet, and facet temperatures from an Abaqus/Standard analysis be written to the output database.

The output request can be repeated as often as necessary to define output for different variables, different cavities, different element sets, different surfaces, etc.

Selecting the Radiation Output Variables

The radiation output variables that can be written to the output database are listed in *Cavity Radiation Variables*.

Input File Usage: **RADIATION OUTPUT*
list of output variables

Selecting the Region of the Model for Which Radiation Output Is Required

You can specify the cavity, element set, or surface for which radiation output is required. Each radiation output request can apply to only one type of region. If you do not specify a region of the model, radiation variables are output for all the cavities in the model.

Input File Usage: Use one of the following options:

**RADIATION OUTPUT*, CAVITY=*cavity_name*
**RADIATION OUTPUT*, ELSET=*element_set_name*
**RADIATION OUTPUT*, SURFACE=*surface_name*

Controlling the Output Frequency

The frequency of radiation output is controlled as described in *Controlling the Frequency of Output to the Output Database*.

Requesting Output

You can request all radiation variables applicable to the current procedure. In this case any additional variables you specify are ignored.

Input File Usage: **RADIATION OUTPUT*, VARIABLE=ALL

Examples of Field and History Output Requests

Products: Abaqus/Standard Abaqus/Explicit

References:

- *About Output*
- **OUTPUT*

Overview

The examples that follow illustrate how to request multiple types of output over multiple steps in both Abaqus/Standard and Abaqus/Explicit.

Abaqus/Standard Example

The input listing below will produce both field and history output for Step 1. Field output will be written every 2 increments. This field output request consists of preselected element variables for the whole model, as well as the variable PEQC. In addition, plastic strains will be written out for element set `SMALL`, and the nodal variables U and RF will be written to the output database for node set `NSMALL`. History output will be written every increment. The variables ALLKE, ALLSE, and ALLWK will be written for the whole model. In addition, ALLPD will be written for element set `SMALL`.

In Step 2 the history output request defined in Step 1 is replaced by a request for the energy variables ALLKE, ALLPD, and ALLSE for element set `SMALL`. The history output request defined in Step 1 is removed. The field output request defined in Step 1 is passed into Step 2 unchanged, but another field output request for element energies at every increment is added.

```
*STEP
*STATIC
...
...
*OUTPUT, FIELD, FREQUENCY=2
*ELEMENT OUTPUT, VARIABLE=PRESELECT
PEQC,
*ELEMENT OUTPUT, ELSET=SMALL
PE,
*NODE OUTPUT, NSET=NSMALL
U, RF
*OUTPUT, HISTORY, FREQUENCY=1
*ENERGY OUTPUT
ALLKE, ALLSE, ALLWK
*ENERGY OUTPUT, ELSET=SMALL
ALLPD
*END STEP
*STEP
*STATIC
...
...
*OUTPUT, HISTORY, OP=REPLACE, FREQUENCY=1
*ENERGY OUTPUT, ELSET=SMALL
ALLKE, ALLPD, ALLSE
*OUTPUT, FIELD, OP=ADD, FREQUENCY=1
```

```
*ELEMENT OUTPUT
ELEN
*END STEP
```

Abaqus/Explicit Example

The input listing below will produce both field and history output for Step 1. Field output will be written at 5 equally spaced intervals, and the time marks will be hit exactly. This field output request consists of preselected element variables for the whole model, as well as the variable PEQC. In addition, plastic strains will be written out for element set `SMALL`, and the nodal variables U and RF will be written to the output database for node set `NSMALL`. History output will be written at a time interval of 0.005. The Abaqus/Explicit time step, DT, will be written, along with the variables ALLKE, ALLSE, and ALLWK for the whole model. The output variables SOAREA and SOF integrated over the surface `CROSS_SECTION1` will be written. The preselected variables SOF and SOM integrated over the surface `CROSS_SECTION2` defined by the integrated output section `SECTION1` will be written in the local coordinate system `LOCALSYSTEM`. In addition, ALLPD will be written for element set `SMALL`.

In Step 2 the history output request defined in Step 1 is replaced by a request for the energy variables ALLKE, ALLPD, and ALLSE for element set `SMALL`. The history output request defined in Step 1 is removed. The field output request defined in Step 1 is passed into Step 2 unchanged, but another field output request for element energies at 10 equally spaced intervals is added.

```
*STEP
*DYNAMIC, EXPLICIT,.1...
...
*OUTPUT, FIELD, NUMBER INTERVAL=5, TIME MARKS=YES
*ELEMENT OUTPUT, VARIABLE=PRESELECT
PEQC,
*ELEMENT OUTPUT, ELSET=SMALL
PE,
*NODE OUTPUT, NSET=NSMALL
U, RF
*OUTPUT, HISTORY, TIME INTERVAL=0.005
*INCREMENTATION OUTPUT
DT
*ENERGY OUTPUT
ALLKE, ALLSE, ALLWK
*ENERGY OUTPUT, ELSET=SMALL
ALLPD
*INTEGRATED OUTPUT, SURFACE=CROSS_SECTION1
SOF, SOAREA
*INTEGRATED OUTPUT SECTION, NAME=SECTION1,
SURFACE=CROSS_SECTION2, ORIENTATION=LOCALSYSTEM
*INTEGRATED OUTPUT, SECTION=SECTION1, VARIABLE=PRESELECT
*END STEP
*STEP
*DYNAMIC, EXPLICIT,.1...
...
*OUTPUT, HISTORY, OP=REPLACE, TIME INTERVAL=0.005
*ENERGY OUTPUT, ELSET=SMALL
ALLKE, ALLPD, ALLSE
*OUTPUT, FIELD, OP=ADD, NUMBER INTERVAL=10
*ELEMENT OUTPUT
ELEN
*END STEP
```

Error Indicator Output

Products: Abaqus/Standard Abaqus/CAE

References:

- *Abaqus/Standard Output Variable Identifiers*
- *About Adaptive Remeshing*
- *Selection of Error Indicators Influencing Adaptive Remeshing*
- **CONTACT OUTPUT*
- **ELEMENT OUTPUT*

Overview

To aid you in understanding the extent and spatial distribution of the discretization error in a finite element solution, Abaqus/Standard provides a set of error indicator output variables.

Error indicator output variables:

- indicate discretization error in a solution quantity (the base solution) and have units of the base solution;
- can be requested with element output or contact output options or as part of an adaptive remeshing rule;
- can be normalized by forms of the base solution to obtain nondimensional, such as percentage, indicators of error;
- can increase your analysis solution time significantly in some cases; and
- are available in Abaqus/Standard but not Abaqus/Explicit.

Warning: Error indicator output variables are approximate and do not represent an accurate or conservative estimate of your solution error. The quality of an error indicator can be particularly poor if your mesh is coarse. The error indicator quality improves as you refine the mesh; however, you should never interpret these variables as indicating what the value of a solution variable would be upon further refinement of the mesh.

Solution Accuracy

The ability of a finite element analysis to make useful predictions of physical behavior depends on many factors, including:

- representation of geometry, material behavior, load history, and various other modeling aspects associated with describing the problem posed;
- spatial and temporal discretization (mesh refinement and incrementation); and
- convergence tolerances.

The primary focus of this section is spatial discretization error. Discussion to help understand and control other potential sources of error appears in *Convergence Criteria for Nonlinear Problems*, *Time Integration Accuracy in Transient Problems*, *Evaluating hyperelastic, hyperfoam and viscoelastic material behavior*, and other portions of the Abaqus documentation. You should perform a detailed study of your analysis methods and assumptions as part of any error assessment.

Spatial Discretization Error

The finite element discretization of a model domain results in an approximation to the exact solution for all but trivial analyses. To aid you in understanding the extent and spatial distribution of the discretization error in a finite element solution, Abaqus/Standard provides a set of error indicator output variables. Ideally, error indicator output variables should be supplemented by other techniques, such as a mesh refinement study, to gain confidence that discretization error is not significantly degrading the ability of the finite element analysis to make useful predictions. In fact, error indicators can help automate a mesh refinement study through the adaptive remeshing functionality of Abaqus/CAE; error indicators variables are used by this functionality to determine where to refine or coarsen a mesh (see *About Adaptive Remeshing*).

Error Indicator and Base Solution Variables Available in Abaqus/Standard

Abaqus error indicator variables provide a measure of the local error resulting from mesh discretization. Each error indicator, c_e, provides an indication of error in a particular base solution variable, c_b. For example, the Mises stress error indicator, MISESERI, provides an indicator of error in the Mises stress variable MISESAVG. *Table 1* shows the available error indicator variables and the corresponding base solution variables.

Table 1: Error indicator variables and their corresponding base solution variables.

Solution Quantity	Error indicator variable (c_e)	Base solution variable (c_b)
Element energy density	ENDENERI	ENDEN
Mises stress	MISESERI	MISESAVG
Contact pressure	CPRESSERI	CPRESS
Contact shear stress	CSHEARERI	CSHEAR
Equivalent plastic strain	PEEQERI	PEEQAVG
Plastic strain	PEERI	PEAVG
Creep strain	CEERI	CEAVG
Heat flux	HFLERI	HFLAVG
Electric flux	EFLERI	EFLAVG
Electric potential gradient	EPGERI	EPGAVG

The algorithms used by Abaqus/CAE to modify mesh seed sizes for the adaptive remeshing capability consider error indicator values and corresponding base solution values together. When you create a remeshing rule and request a particular error indicator, Abaqus automatically writes the error indicator and corresponding base solution variable to the output database.

Input File Usage: **OUTPUT*, FIELD
**ELEMENT OUTPUT*, ELSET=*ElsetName*
**CONTACT OUTPUT*

Abaqus/CAE Usage: Step module: **Output->Field Output Request**

Or, if you use the following option to specify an adaptive remeshing rule, the associated error indicator and base solution output will occur by default:

Mesh module: **Create Remeshing Rule**: **Step and Indicator**

Effect of Error Indicator Output Requests on Solution Time

Abaqus/Standard determines error indicator variables based on the difference between a smoothed and unsmoothed distribution of the base solution, using a smoothing technique such as the patch recovery technique of Zienkiewicz and Zhu, (1987). The smoothing calculations occasionally noticeably increase analysis time. If you find that adding an error indicator output request significantly increases analysis time, strategies for reducing this effect include reducing the output frequency and limiting the output request to a particular region of interest. Computations for most error indicator variables only occur just prior to writing the error indicator variable to the output database, so reducing the output frequency will tend to reduce the computation time; however this is not the case for the element energy density error indicator, because contributions to this error indicator are accumulated each increment regardless of whether this error indicator is output for a given increment.

Additional Considerations for Extent of Output Requests for Element Error Indicator Variables

When you request element error indicator output, the request should only apply to elements supported for error indicator output.

The patch recovery technique used to compute element error indicator variables assumes that the solution should be continuous over the element set specified. Abaqus/Standard confirms that your error indicator output specification is consistent with this assumption by checking section property references within the error indicator domain and issues a warning message if the elements in the provided element set refer to distinct section definitions. You can safely ignore this warning if the sections are identical in their properties.

Interpreting Error Indicator Output

When interpreting error indicator output, you should remember that the error indicators are approximate measures of the local error in the base solution and are, themselves, subject to discretization error. The accuracy of the error estimates tends to improve as the mesh is refined. Each error indicator variable has the same units has the corresponding base solution variable, which facilitates comparison of local estimates of the error magnitude with local estimates of the base solution.

Regions of Interest of a Base Solution and Corresponding Error Indicator

Viewing contour plots of a base solution variable and corresponding error indicator variable side-by-side can provide a useful perspective on the solution accuracy. For example, if the base solution is expressed in units of stress, the corresponding error indicator is also expressed in units of stress. *Figure 1* shows contour plots of CPRESS and CPRESSERI for an analysis of a sphere pressed into a rigid plate. These plots can be interpreted as follows:

- The contact pressure solution is quite accurate near the center of the active contact region, where the contact pressure is largest, because the error indicator is a small fraction of the base solution in that region.
- The contact pressure solution is less accurate near the perimeter of the active contact region, where local variations in the contact pressure solution are largest (but the contact pressure is significantly less than the maximum value), because the error indicator is quite large compared to the base solution in that region.

The analyst may judge that the level of mesh refinement is adequate if the maximum contact pressure is of primary interest in such a case. Local mesh refinement would be needed to accurately predict the maximum contact pressure if the active contact region was significantly smaller than that shown in *Figure 1*.

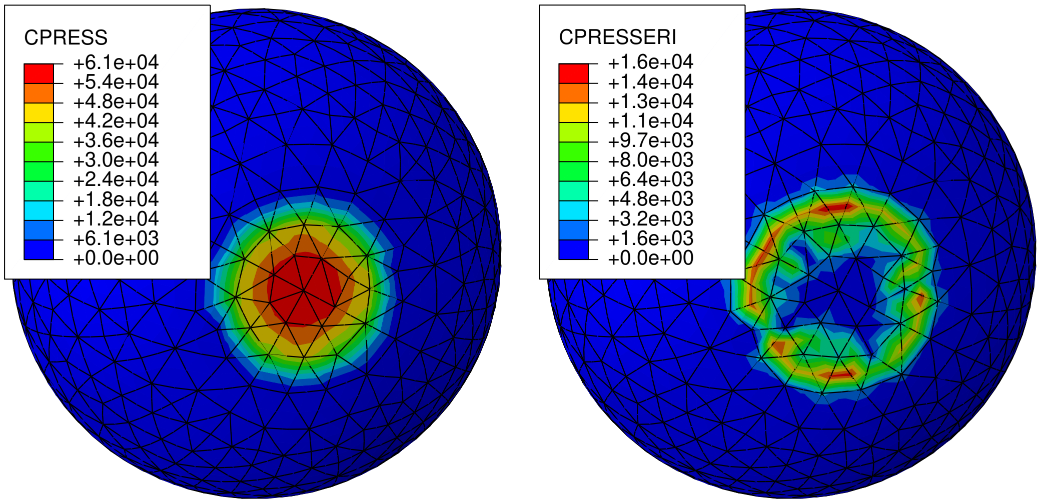


Figure 1: Contour plots of CPRESS and CPRESSERI for contact between a deformable sphere and a rigid plate.

An error indicator tends to give a crude, non-conservative approximation of the deviation from the exact solution if the mesh is coarse relative to local solution variations or the exact solution to the problem posed involves a stress singularity. The following qualitative interpretations of error indicator results exceeding approximately 10% of base solution results are often appropriate:

- "Significant potential for solution inaccuracy exists in this region."
- "The mesh may be too coarse to give a good estimate of solution error in this region."
- "Perhaps a stress singularity exists at this corner."

Calculating Normalized Measures of Solution Error

You can use corresponding error indicator and base solution variables, c_e and c_b, respectively, to compute a field of local, normalized error indicators:

$$\hat{c}_e = \frac{c_e}{c_b},$$

where $\hat{c}_e$ is a normalized error measure. For example,

$$\left(\frac{\text{MISESERI}}{\text{MISESAVG}}\right) \times 100$$

provides a percentage form of the Mises stress-based error indicator; however this normalized error measure may not be particularly useful, because it:

- will tend to draw attention to regions where base solution values are small, which typically are not critical regions of a design; and
- will have divide-by-zero issues where the base solution value is zero.

Other normalization approaches, such as normalizing based on a global norm of the base solution variable or a constant value that you choose (such as the maximum value of the base solution allowed in a design), may be more effective.

Normalized forms of an error indicator are not available directly through the error indicator output variables; however, you can calculate normalized measures using the Visualization module of Abaqus/CAE (Abaqus/Viewer) to operate on field output data. For more information, see *Building valid field output expressions*. Alternatively, you can use the Abaqus Scripting Interface to read the error indicator and the base solution from the output database and calculate normalized forms. For more information, see *Using the Abaqus Scripting Interface to access an output database*.

Limitations

Only the following element types are supported for error indicator computations:

- Planar continuum triangles and quadrilaterals
- Shell triangles and quadrilaterals
- Tetrahedrals
- Hexahedrals

Elements with variable nodes are not supported.

Error indicator output is not supported in the following cases:

- Import analysis
- Restart analysis
- Post output analysis
- Map solution analysis
- Symmetric model generation analysis

References

- Zienkiewicz, O. C., and J. Z. Zhu, “A Simple Error Estimator and Adaptive Procedure for Practical Engineering Analysis,” *International Journal for Numerical Methods in Engineering*, vol. 24, pp. 337–357, 1987.

Output Variables

In this section:

Using Abaqus/Standard Output Variable Identifiers

In this section:

Abaqus/Standard Output Variable Identifiers

Products: Abaqus/Standard

References:

- *About Output*
- *Output to the Data and Results Files*
- *Output to the Output Database*
- *Loads*
- *Abaqus/Standard User Subroutines*

Overview

The tables in the following sections list all of the output variables that are available in Abaqus/Standard.

These output variables can be requested as either field- or history-type output to the output database (`.odb`) file (see *Output to the Output Database*) or for output to the results (`.fil`) and data (`.dat`) files (see *Output to the Data and Results Files*). In general, output variables that can be requested as field- or history-type output to an output database in ODB format can also be requested as output in SIM format (see *The Output Database*). As noted specifically in the tables, a few of the output variables are written only to the output database and restart (`.res`) files (they are not available for output to the data or results files). These variables can be accessed only in the Visualization module of Abaqus/CAE (Abaqus/Viewer).

Notation Used in the Output Variable Descriptions

The entries “Field”, “History”, “.fil”, and “.dat” in the output variable's description indicate the availability of the output variable. “Field” refers to a field-type output selection to the output database, “History” refers to a history-type output selection to the output database, “.fil” refers to a results file output selection, and “.dat” refers to a data file output selection. The output variable can be written to the respective file if the word “yes” appears after the category name; “no” means that the variable is not available to that file.

If the word “automatic” appears after a category name, the variable cannot be requested by name; it is written to the respective files according to the conditions specified in the text.

Requesting Output of Components

Variable identifiers of the form *ABCn* can be used with $n = 1, 2, 3, \ldots$ (*ABC*1, *ABC*2, …), where the highest value of *n* is determined by the type of variable. Similarly, variable identifiers of the form $DEFij$ can be used for the ranges of *i* and *j* indicated (*DEF*11, *DEF*12, ...).

Individual components cannot be requested in the results (`.fil`) file. For postprocessing of a particular component of a variable, request file output for all components of the variable. Output for individual variables can be requested during postprocessing.

Individual components of variables can be requested as history-type output in the output database for *X*–*Y* plotting in Abaqus/CAE. Individual component requests to the output database are not available for field-type output, with the exception of state, field, and user-defined variables (SDV*n*, ESDV*n*, FV*n*, and UVARM*n*). If a particular component is desired for contouring in Abaqus/CAE, request field output of the generic variable (e.g., S for stress). Output for individual components of field output can be requested within the Visualization module of Abaqus/CAE.

Direction Definitions

The direction definitions depend on the variable type.

Direction Definitions for Element Variables

For components of stress, strain, and other tensor quantities 1, 2, and 3 refer to the directions in an orthogonal coordinate system. These directions are global directions for solid elements, surface directions for shell and membrane elements, and axial and transverse directions for beam elements. For finite-membrane-strain shell elements, membrane elements, and continuum elements associated with a local orientation (see *Orientations*), the local output directions rotate with the average rotation of the element (integral with respect to time of the spin—see *Stress rates*). Tensor components in these cases are output in the rotating local directions.

In some cases the local output directions may differ from one integration point to the next within an element. Abaqus/Standard does not take this variation into account when extrapolating output variables to the nodes, which affects output such as element quantities averaged at the nodes or contour plots of individual tensor components. Invariant quantities at the integration points will not be influenced by the local output directions.

You can control writing the local directions to the output database file or to the results file (see *Specifying the Directions for Element Output* and *Output of Local Directions to the Results File*). By default, the local directions are written to the output database for all frames that include element field output. The local (material) directions (averaged at the nodes) can be visualized in Abaqus/CAE by selecting **Plot->Material Orientations** in the Visualization module. The directions can be printed to the data file by using user subroutine *UVARM*.

Direction Definitions for Equivalent Rigid Body Variables

For all equivalent rigid body variables 1, 2, and 3 refer to global directions.

Direction Definitions for Nodal Variables

For nodal variables 1, 2, and 3 are global directions (1=X, 2=Y, and 3=Z; or for axisymmetric elements, 1=r and 2=z). If a local coordinate system is defined at a node (see *Transformed Coordinate Systems*), you can specify whether output to the data or results file of vector-valued quantities at these nodes is in the local or global system (see *Specifying the Directions for Nodal Output*). By default, nodal output is written to the data file in the local system, whereas it is written to the results file in the global system (since this is more convenient for postprocessing).

If nodal field output is requested for a node that has a local coordinate system defined, a quaternion representing the rotation from the global directions is written to the output database. Abaqus/CAE automatically uses this quaternion to transform the nodal results into the local directions. Nodal history data written to the output database are always stored in the global directions.

Direction Definitions for Integrated Variables

For components of total force, total moment, and similar variables obtained through integration over a surface, the directions 1, 2, and 3 refer to directions in an orthogonal coordinate system. A fixed global coordinate system is used if the surface is specified directly for the integrated output request. If the surface is identified by an integrated output section definition (see *Integrated Output Section Definition*) that is associated with the integrated output request, a local coordinate system in the initial configuration can be specified and can translate or rotate with the deformation.

Distributed Load Output

You need to be aware of limitations that may be encountered when distributed load output is requested.

Distributed Load Output and User Subroutines

Output can be requested for many of the distributed loads discussed in *Loads*. However, contributions to these loads defined through user subroutines (see *Abaqus/Standard User Subroutines*) are not displayed, except for the variables FILMCOEF and SINKTEMP.

Distributed Load Output with Modal Procedures

For modal procedures only the magnitude of the load is written to the output database.

Strain Output

The total strain E is composed of the elastic strain EE, the inelastic strain IE, and the thermal strain THE. The inelastic strain IE consists of the plastic strain PE and the creep strain CE.

For geometrically nonlinear analysis Abaqus/Standard makes it possible to output different strain measures as well as elastic and various inelastic strains. The various total strain measures (integrated strain measure E, nominal strain measure NE, and logarithmic strain measure LE) are described in *Conventions*. The default strain measure for output to the data (`.dat`) and results (`.fil`) files is E. However, for geometrically nonlinear analysis using element formulations that support finite strains, E is not available for output to the output database (`.odb`) file, and LE is the default strain measure.

Temperature Output

In Abaqus temperature can either be a field variable (stress analysis, mass diffusion, …) or a degree of freedom (heat transfer analysis, fully coupled temperature-displacement analysis, …). For any analysis that involves temperature, you can request the temperature either at nodes (variable NT) or in elements (variable TEMP). If element temperature output is requested at the nodes, the integration point values are extrapolated and, if requested, averaged. These extrapolated values are generally not as accurate as the nodal temperatures themselves. An exception to this is adiabatic analysis, in which the element temperatures change due to plastic heat generation but the nodal temperatures are not updated. In that case the current nodal temperatures are obtained only if element temperature output is requested at the nodes.

For continuum elements there is only one temperature value per node (NT11). For shells and beams more than one temperature is available for each node (NT11, NT12, …) since a temperature gradient can exist through the thickness of a shell or across the cross-section of a beam. In general, variables NT12, NT13, etc. contain temperature values. However, when temperature is defined by specifying temperature gradients, nodal temperatures for a given section point can be obtained only by using the variable TEMP. See *Specifying Temperature and Field Variables* and *Specifying Temperature and Field Variables* for discussions on specifying temperatures in beams and shells.

Principal Value Output

Output of the principal values can be requested for stresses, strains, and other material tensors. Either all principal values or the minimum, maximum, or intermediate values can be obtained. All principal values of tensor *ABC* are obtained with the request *ABC*P. The minimum, intermediate, and maximum principal values are obtained with the requests *ABC*P1, *ABC*P2, and *ABC*P3.

For three-dimensional, (generalized) plane strain, and axisymmetric elements all three principal values are obtained. For plane stress, membrane, and shell elements, the out-of-plane principal value cannot be requested for history-type output. For field-type output, Abaqus/CAE always assumes the out-of-plane principal value as zero, including when computing the **Max. Principal**, **Mid. Principal**, and **Min. Principle** values. Principal values cannot be obtained for truss elements or for any beam elements other than the three-dimensional beam elements with torsional shear stresses.

If a principal value or an invariant is requested for field-type output, the output request is replaced with an output request for the components of the corresponding tensor. Abaqus/CAE calculates all principal values and invariants from these components. If a principal value is desired as history-type output, it must be explicitly requested since Abaqus/CAE does no calculations on history data.

Tensor Output

Tensor variables that are written to the output database as field-type output are written as components in either the default directions defined by the convention given in *Orientations* (global directions for solid elements, surface directions for shell and membrane elements, and axial and transverse directions for beam elements), or the user-defined local system. Abaqus/CAE calculates all principal values and invariants from these components. See *Writing field output data*, for a description of the different types of tensor variables.

For plane stress, membrane, and shell elements, only the in-plane tensor components (11, 22, and 12 components) are stored by Abaqus/Standard. The out-of-plane direct component for stress (S33) is reported as zero to the output database as expected, and the out-of-plane component of strain (E33) is reported as zero even though it is not. This is because the thickness direction is computed based on section properties rather than at the material level. The out-of-plane components can be requested for field-type output and cannot be requested for history-type output. The out-of-plane stress components are not reported to the data (`.dat`) file or to the results (`.fil`) file.

For three-dimensional beam elements with torsional shear stresses, only the axial and the torsional components (the 11 and 12 components) are stored by Abaqus/Standard. The other direct component (the 22 component) is reported as zero for field-type output and cannot be requested for history-type output.

The components for tensor variables are written to the output database in single precision. Therefore, a small amount of precision roundoff error may occur when calculating the variables' principal values. Such roundoff error may be observed, for example, when analytically zero values are calculated as relatively small nonzero values.

Element Integration Point Variables

The output variables listed below are available in Abaqus/Standard.

References:

- *Element Output*
- *Writing Element Output to the Output Database*

Tensors and Associated Principal Values and Invariants

S
Field: yes History: yes .fil: yes .dat: yes

All stress components.

S*ij*
Field: no History: yes .fil: no .dat: yes

ij-component of stress ($i \leq j \leq 3$).

SP
Field: yes History: yes .fil: yes .dat: yes

All principal stresses.

SP*n*
Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal stresses (SP1≤SP2≤SP3).

SINV
Field: yes History: yes .fil: yes .dat: yes

All stress invariant components (MISES, TRESC, PRESS, INV3). For field output SINV is converted to a request for the generic variable S.

S_MISES
Field: yes History: yes .fil: no .dat: no

Signed von Mises equivalent stress. The sign of this quantity is the same as the sign of the trace of the stress tensor.

MISES
Field: no History: yes .fil: no .dat: yes

Mises equivalent stress, defined as

$$q = \sqrt{\tfrac{3}{2}\mathbf{S} : \mathbf{S}},$$

where $\mathbf{S}$ is the deviatoric stress tensor, defined as $\mathbf{S} = \boldsymbol{\sigma} + p\,\mathbf{I},$ where $\boldsymbol{\sigma}$ is the stress, p is the equivalent pressure stress (defined below), and $\mathbf{I}$ is a unit matrix. In index notation

$$q = \sqrt{\tfrac{3}{2}S_{ij}S_{ij}},$$

where $S_{ij} = \sigma_{ij} + p\,\delta_{ij}$, $p = -\frac{1}{3}\sigma_{ii}$, and δ_{ij} is the Kronecker delta.

MISESMAX
Field: yes History: no .fil: no .dat: no

Maximum Mises stress among all the section points. For a shell element it represents the maximum Mises value among all the section points in the layer, for a beam element it is the maximum Mises stress among all the section points in the cross-section, and for a solid element it represents the Mises stress at the integration points.

MISESONLY Field: yes History: no .fil: no .dat: no

Mises equivalent stress. When MISESONLY is used instead of MISES, the stress components are not written to the output database; consequently, the size of the database is reduced.

TRESC Field: no History: yes .fil: no .dat: yes

Tresca equivalent stress, defined as the maximum difference between principal stresses.

PRESS Field: no History: yes .fil: no .dat: yes

Equivalent pressure stress, defined as

$p = -\frac{1}{3}\,\mathrm{trace}\,(\boldsymbol{\sigma}) = -\frac{1}{3}\sigma_{ii}.$

PRESSONLY Field: yes History: no .fil: no .dat: no

Equivalent pressure stress. When PRESSONLY is used instead of PRESS, the stress components are not written to the output database; consequently, the size of the database is reduced.

INV3 Field: no History: yes .fil: no .dat: yes

Third stress invariant, defined as

$$r = \left(\frac{9}{2}\mathbf{S}\cdot\mathbf{S}:\mathbf{S}\right)^{1/3} = \left(\frac{9}{2}S_{ij}S_{jk}S_{ki}\right)^{1/3},$$

where $\mathbf{S}$ is the deviatoric stress defined in the context of Mises equivalent stress, above.

TRIAX Field: yes History: yes .fil: no .dat: no

Stress triaxiality, $\eta = -p/q$.

YIELDS Field: yes History: yes .fil: no .dat: no

Yield stress, σ^0, available for Mises, Johnson-Cook, and Hill plasticity material models.

ALPHA Field: yes History: yes .fil: yes .dat: yes

All total kinematic hardening shift tensor components.

ALPHA*ij* Field: no History: yes .fil: no .dat: yes

ij-component of the total shift tensor ($i \le j \le 3$).

ALPHA*k* Field: yes History: yes .fil: no .dat: no

All k^{th} kinematic hardening shift tensor components ($1 \le k \le 10$).

ALPHA*k_ij* Field: no History: yes .fil: no .dat: no

ij-component of the k^{th} kinematic hardening shift tensor ($i \le j \le 3$ and $1 \le k \le 10$).

ALPHAN Field: yes History: yes .fil: no .dat: no

All tensor components of all the kinematic hardening shift tensors, except the total shift tensor, ALPHA.

ALPHAP Field: yes History: yes .fil: yes .dat: yes

All principal values of the total shift tensor.

ALPHAP*n* Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal values of the total shift tensor (ALPHAP1≤ALPHAP2≤ALPHAP3).

SNET*k* Field: yes History: yes .fil: no .dat: no

All stress components in the k^{th} network ($0 \leq k \leq 10$). Available only for the parallel rheological framework.

SNET*k_ij* Field: no History: yes .fil: no .dat: no

ij-component of stress in the k^{th} network ($i \leq j \leq 3$ and $0 \leq k \leq 10$). Available only for the parallel rheological framework.

E Field: yes History: yes .fil: yes .dat: yes

All strain components. For geometrically nonlinear analysis using element formulations that support finite strains, E is not available for output to the output database (`.odb`) file.

E*ij* Field: no History: yes .fil: no .dat: yes

ij-component of strain ($i \leq j \leq 3$).

EP Field: yes History: yes .fil: yes .dat: yes

All principal strains.

EP*n* Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal strains (EP1≤EP2≤EP3).

NE Field: yes History: yes .fil: yes .dat: yes

All nominal strain components.

NE*ij* Field: no History: yes .fil: no .dat: yes

ij-component of nominal strain ($i \leq j \leq 3$).

NEP Field: yes History: yes .fil: yes .dat: yes

All principal nominal strains.

NEP*n* Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal nominal strains (NEP1≤NEP2≤NEP3).

LE Field: yes History: yes .fil: yes .dat: yes

All logarithmic strain components. For geometrical nonlinear analysis using element formulations that support finite strains, LE is the default strain measure for output to the output database (`.odb`) file.

LE*ij* Field: no History: yes .fil: no .dat: yes

ij-component of logarithmic strain ($i \leq j \leq 3$).

LEP Field: yes History: yes .fil: yes .dat: yes

All principal logarithmic strains.

LEP*n* Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal logarithmic strains (LEP1≤LEP2≤LEP3).

ER Field: yes History: yes .fil: yes .dat: yes

All mechanical strain rate components.

ER*ij* Field: no History: yes .fil: no .dat: yes

ij-component of strain rate ($i \le j \le 3$).

ERP Field: yes History: yes .fil: yes .dat: yes

All principal mechanical strain rates.

ERP*n* Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal mechanical strain rates (ERP1≤ERP2≤ERP3).

DG Field: no History: no .fil: yes .dat: yes

All components of the total deformation gradient. Available only for hyperelasticity, hyperfoam, and material models defined in user subroutine *UMAT*. For fully integrated first-order quadrilaterals and hexahedra, the selectively reduced integration technique is used. A modified deformation gradient is output for these elements.

DG*ij* Field: no History: no .fil: no .dat: yes

ij-component of the total deformation gradient ($i, j \le 3$).

DGP Field: no History: no .fil: yes .dat: yes

Principal stretches.

DGP*n* Field: no History: no .fil: no .dat: yes

Minimum, intermediate, and maximum values of principal stretches (DGP1≤DGP2≤DGP3).

EE Field: yes History: yes .fil: yes .dat: yes

All elastic strain components.

EE*ij* Field: no History: yes .fil: no .dat: yes

ij-component of elastic strain ($i \le j \le 3$).

EEP Field: yes History: yes .fil: yes .dat: yes

All principal elastic strains.

EEP*n* Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal elastic strains (EEP1≤EEP2≤EEP3).

IE Field: yes History: yes .fil: yes .dat: yes

All inelastic strain components.

IE*ij* Field: no History: yes .fil: no .dat: yes

ij-component of inelastic strain ($i \le j \le 3$).

IEP Field: yes History: yes .fil: yes .dat: yes

All principal inelastic strains.

IEP*n*
: Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal inelastic strains (IEP1≤IEP2≤IEP3).

THE
: Field: yes History: yes .fil: yes .dat: yes

All thermal strain components.

THE*ij*
: Field: no History: yes .fil: no .dat: yes

ij-component of thermal strain ($i \leq j \leq 3$).

THEFL
: Field: yes History: yes .fil: no .dat: no

Thermal strain in the pore fluid in a porous medium.

THEP
: Field: yes History: yes .fil: yes .dat: yes

All principal thermal strains.

THEP*n*
: Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal thermal strains (THEP1≤THEP2≤THEP3).

PE
: Field: yes History: yes .fil: yes .dat: yes

All plastic strain components. This identifier also provides PEEQ, a yes/no flag telling if the material is currently yielding or not (AC YIELD: "actively yielding"; that is, the plastic strain changed during the increment), and PEMAG when PE is requested for the data or results files. When PE is requested for field output to the output database, PEEQ is also provided.

PE*ij*
: Field: no History: yes .fil: no .dat: yes

ij-component of plastic strain ($i \leq j \leq 3$).

PEEQ
: Field: yes History: yes .fil: no .dat: yes

Equivalent plastic strain. This identifier also provides a yes/no flag (1/0 on the output database) telling if the material is currently yielding or not (AC YIELD: "actively yielding"; that is, the plastic strain changed during the increment).

The equivalent plastic strain is defined as $\bar{\varepsilon}^{pl}\big|_0 + \int_0^t \dot{\bar{\varepsilon}}^{pl} dt$, where $\bar{\varepsilon}^{pl}\big|_0$ is the initial equivalent plastic strain.

The definition of $\dot{\bar{\varepsilon}}^{pl}$ depends on the material model. For classical metal (Mises) plasticity

$\dot{\bar{\varepsilon}}^{pl} = \sqrt{\frac{2}{3}\dot{\boldsymbol{\varepsilon}}^{pl} : \dot{\boldsymbol{\varepsilon}}^{pl}}$. For other plasticity models, see the appropriate section in *Abaqus Materials Guide*.

When plasticity occurs in the thickness direction to a gasket element whose plastic behavior is specified as part of a gasket behavior definition, PEEQ is PE11.

PEEQMAX
: Field: yes History: no .fil: no .dat: no

Maximum equivalent plastic strain, PEEQ, among all the section points. For a shell element it represents the maximum PEEQ value among all the section points in the layer, for a beam element it is the maximum PEEQ among all the section points in the cross-section, and for a solid element it represents the PEEQ at the integration points.

PEEQT Field: yes History: yes .fil: yes .dat: yes

Equivalent plastic strain in uniaxial tension for cast iron, Mohr-Coulomb tension cutoff, and concrete damaged plasticity, which is defined as $\int \dot{\bar{\varepsilon}}_t^{pl} dt$. This identifier also provides a yes/no flag (1/0 on the output database) telling if the material is currently yielding or not (AC YIELDT: "actively yielding"; that is, the plastic strain changed during the increment).

PEMAG Field: yes History: yes .fil: no .dat: yes

Plastic strain magnitude, defined as $\sqrt{\frac{2}{3}\boldsymbol{\varepsilon}^{pl} : \boldsymbol{\varepsilon}^{pl}}$.

For most materials, PEEQ and PEMAG are equal only for proportional loading. When plasticity occurs in the thickness direction to a gasket element whose plastic behavior is specified as part of a gasket behavior definition, PEMAG is PE11.

PEP Field: yes History: yes .fil: yes .dat: yes

All principal plastic strains.

PEP*n* Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal plastic strains (PEP1≤PEP2≤PEP3).

CE Field: yes History: yes .fil: yes .dat: yes

All creep strain components. This identifier also provides CEEQ, CESW, and CEMAG when CE is requested for the data or results files.

CE*ij* Field: no History: yes .fil: no .dat: yes

ij-component of creep strain ($i \leq j \leq 3$).

CEEQ Field: yes History: yes .fil: no .dat: yes

Equivalent creep strain, defined as $\int_0^t \dot{\bar{\varepsilon}}^{cr} dt$.

The definition of $\dot{\bar{\varepsilon}}^{cr}$ depends on the material model. For classical metal (Mises) creep $\dot{\bar{\varepsilon}}^{cr} = \sqrt{\frac{2}{3}\dot{\boldsymbol{\varepsilon}}^{cr} : \dot{\boldsymbol{\varepsilon}}^{cr}}$. For other creep models, see the appropriate section in *Abaqus Materials Guide*.

When creep occurs in the thickness direction to a gasket element whose creep behavior is specified as part of a gasket behavior definition, CEEQ is CE11.

CESW Field: yes History: yes .fil: no .dat: yes

Magnitude of swelling strain.

For cap creep CESW gives the equivalent creep strain produced by the consolidation creep mechanism, defined as $\int \frac{\boldsymbol{\sigma}:d\boldsymbol{\varepsilon}^{cr}}{\bar{p}}$, where $\bar{p}$ is the equivalent creep pressure, $\bar{p} = \left(R^2\ q^2 + p\ \ (p - p_a)\right) / G_c^{cr}$.

CEMAG Field: yes History: yes .fil: no .dat: yes

Magnitude of creep strain (defined by the same formula given above for PEMAG, applied to the creep strains).

CEP
Field: yes History: yes .fil: yes .dat: yes

All principal creep strains.

CEP*n*
Field: no History: yes .fil: no .dat: yes

Minimum, intermediate, and maximum principal creep strains (CEP1≤CEP2≤CEP3).

EEIG
Field: yes History: yes .fil: no .dat: no

All eigenstrain components.

EEIG*ij*
Field: no History: yes .fil: no .dat: no

ij-component of eigenstrain ($i \leq j \leq 3$).

MAXPSCRT
Field: yes History: yes .fil: no .dat: no

Maximum principal stress initiation criterion.

LODE
Field: yes History: yes .fil: no .dat: no

Lode angle term, $\xi = \cos(3\Theta)$, where Θ is the Lode angle.

SDEFRES
Field: yes History: yes .fil: no .dat: no

Deformation resistance in Anand's creep model.

Additional Element Stresses

CS11
Field: yes History: yes .fil: yes .dat: yes

Average contact pressure for link and three-dimensional line gasket elements. Available only if the gasket contact area is specified; see *Defining the Contact Area for Average Contact Pressure Output*.

TSHR
Field: yes History: yes .fil: yes .dat: yes

All transverse shear stress components. Available only for thick shell elements such as S3R, S4R, S8R, and S8RT. Contouring of this variable is supported in the Visualization module of Abaqus/CAE.

TSHR*i*3
Field: no History: yes .fil: no .dat: yes

$i3$-component of transverse shear stress ($i = 1, 2$). Available only for thick shell elements such as S3R, S4R, S8R, and S8RT.

CTSHR
Field: yes History: yes .fil: no .dat: yes

Transverse shear stress components for stacked continuum shell elements. Available only for SC6R and SC8R elements. Contouring of this variable is supported in the Visualization module of Abaqus/CAE.

CTSHR*i*3
Field: no History: yes .fil: no .dat: yes

$i3$-component of transverse shear stress ($i = 1, 2$). Available only for SC6R and SC8R elements.

SS
Field: no History: no .fil: yes .dat: yes

All substresses. Available only for ITS elements.

SS*n* Field: no History: no .fil: no .dat: yes

*n*th substress ($n = 1, 2$). Available only for ITS elements.

ORITENS Field: yes History: no .fil: no .dat: yes

All orientation tensor components. Available only for elements with multiscale material or linear orthotropic elastic material with fiber dispersion and only if the orientation tensor is specified with a distribution (*Distribution Definition*).

Vibration and Acoustic Quantities

INTEN Field: yes History: yes .fil: no .dat: no

Vibration intensity. Available only for the steady-state dynamics procedure. For real-only steady-state dynamics analyses, the intensity is a pure imaginary vector, but it is stored as real on the output database. Available for structural, solid, and acoustic elements and for rebar.

ACV Field: yes History: yes .fil: no .dat: no

Acoustic particle velocity. Available only if the steady-state dynamic procedure is used, and available only for acoustic and poroelastic acoustic finite elements.

ACV*n* Field: no History: yes .fil: no .dat: no

Component *n* of the acoustic particle velocity vector ($n = 1, 2, 3$). Available only if the steady-state dynamic procedure is used, and available only for acoustic and poroelastic acoustic finite elements.

GRADP Field: yes History: yes .fil: no .dat: no

Acoustic pressure gradient. Available only if the steady-state dynamic direct-solution or natural frequency extraction procedure is used, and available only for acoustic and poroelastic acoustic finite elements.

GRADP*n* Field: yes History: yes .fil: no .dat: no

Component *n* of the acoustic pressure gradient ($n = 1, 2, 3$). Available only if the steady-state dynamic direct-solution or natural frequency extraction procedure is used, and available only for acoustic and poroelastic acoustic finite elements.

ACDISP Field: yes History: yes .fil: no .dat: yes

Acoustic fluid displacement. Available only if the direct steady-state dynamic procedure is used, and available only for poroelastic acoustic finite elements.

ACDISP*n* Field: yes History: yes .fil: no .dat: yes

Component *n* of the acoustic fluid displacement ($n = 1, 2, 3$). Available only if the direct steady-state dynamic procedure is used, and available only for poroelastic acoustic finite elements.

ACADMIT Field: yes History: yes .fil: no .dat: yes

Acoustic admittance (a reciprocal of impedance). Available only if the direct steady-state dynamic procedure is used, and available only for poroelastic acoustic finite elements.

ACADMIT*n* Field: yes History: yes .fil: no .dat: yes

Component *n* of the acoustic admittance (a reciprocal of impedance) ($n = 1, 2, 3$). Available only if the direct steady-state dynamic procedure is used, and available only for poroelastic acoustic finite elements.

Energy Densities

In steady-state dynamics all energy quantities are net per-cycle values, unless otherwise noted (see *Energy balance*).

ENER Field: yes History: yes .fil: yes .dat: yes

All energy densities. None of the energy densities are available in mode-based procedures; a limited number of them are available for direct-solution steady-state dynamic and subspace-based steady-state dynamic analyses.

SENER Field: yes History: yes .fil: no .dat: yes

Elastic strain energy density (with respect to current volume). When the Mullins effect is modeled with hyperelastic materials, this quantity represents only the recoverable part of energy per unit volume. This is the only energy density available in the data file for eigenvalue extraction procedures; to obtain this quantity for eigenvalue extraction procedures in the results file or as field output in the output database, request ENER. In steady-state dynamic analysis this is the cyclic mean value.

PENER Field: yes History: yes .fil: no .dat: yes

Energy dissipated by rate-independent and rate-dependent plasticity, per unit volume. For superelastic materials, this variable also includes recoverable phase-transformation energy. This output variable is not available for steady-state dynamic analysis.

CENER Field: yes History: yes .fil: no .dat: yes

Energy dissipated by creep, swelling, viscoelasticity, and energy associated with viscous regularization for cohesive elements, per unit volume. Not available for steady-state dynamic analysis.

VENER Field: yes History: yes .fil: no .dat: yes

Energy dissipated by viscous effects (except those from viscoelasticity and static dissipation), per unit volume.

EENER Field: yes History: yes .fil: no .dat: yes

Electrostatic energy density. Not available for steady-state dynamic analysis.

JENER Field: yes History: yes .fil: no .dat: yes

Electrical energy dissipated as a result of the flow of current, per unit volume. Not available for steady-state dynamic analysis.

DMENER Field: yes History: yes .fil: no .dat: yes

Energy dissipated by damage, per unit volume. Not available for steady-state dynamic analysis.

State, Field, and User-Defined Output Variables

SDV Field: yes History: yes .fil: yes .dat: yes

Solution-dependent state variables.

SDVn Field: yes History: yes .fil: no .dat: yes

Solution-dependent state variable *n*.

TEMP Field: yes History: yes .fil: yes .dat: yes

Temperature.

POR Field: yes History: yes .fil: no .dat: yes

Pore fluid pressure.

FV Field: yes History: yes .fil: yes .dat: yes

Predefined field variables, including those imported using the FV_i co-simulation field ID.

FV*n* Field: yes History: yes .fil: no .dat: yes

Predefined field variable *n*.

FVE Field: yes History: yes .fil: no .dat: no

All components of all field expansion strain tensors.

FVE*n* Field: yes History: yes .fil: no .dat: no

All components for the field expansion strain tensor due to field variable number *n*.

FVE*n_ij* Field: no History: yes .fil: no .dat: no

i-j component of the field expansion strain tensor due to field variable number *n*.

FVEFL Field: yes History: yes .fil: no .dat: no

All field expansion strains in the pore fluid in a porous medium.

FVEFL*n* Field: yes History: yes .fil: no .dat: no

Field expansion strain in the pore fluid in a porous medium due to field variable number *n*.

MFR Field: yes History: yes .fil: yes .dat: yes

Predefined mass flow rates.

MFR*n* Field: no History: yes .fil: no .dat: yes

Component *n* of predefined mass flow rate ($n = 1, 2, 3$).

UVARM Field: yes History: yes .fil: yes .dat: yes

User-defined output variables.

UVARM*n* Field: yes History: yes .fil: no .dat: yes

User-defined output variable *n*.

Composite Failure Measures

CFAILURE Field: yes History: yes .fil: yes .dat: yes

All failure measure components.

MSTRS Field: yes History: yes .fil: no .dat: yes

Maximum stress theory failure measure.

TSAIH Field: yes History: yes .fil: no .dat: yes

Tsai-Hill theory failure measure.

TSAIW Field: yes History: yes .fil: no .dat: yes

Tsai-Wu theory failure measure.

AZZIT Field: yes History: yes .fil: no .dat: yes

Azzi-Tsai-Hill theory failure measure.

MSTRN Field: yes History: yes .fil: no .dat: yes

Maximum strain theory failure measure.

Fluid Link Quantities

MFL Field: yes History: yes .fil: yes .dat: yes

Current value of the mass flow rate.

MFLT Field: yes History: yes .fil: yes .dat: yes

Current value of the total mass flow.

Fluid Pipe Element Quantities

FPMFL Field: yes History: yes .fil: no .dat: no

Current value of the mass flow rate.

FPFLVEL Field: yes History: yes .fil: no .dat: no

Current velocity of the fluid flowing through the pipe.

FPDPRESS Field: yes History: yes .fil: no .dat: no

Current pressure drop across the element.

Fracture Mechanics Quantities

JK Field: yes History: yes .fil: yes .dat: yes

J-integral, stress intensity factors. Available only for line spring elements. Output is in the following order for LS3S elements: J, K, J^{el}, and J^{pl}. Output is in the following order for LS6 elements: J, J^{el}, J^{pl}, K_I, K_{II}, and K_{III}.

Concrete Cracking and Additional Plasticity

CRACK Field: no History: no .fil: yes .dat: yes

Unit normal to cracks in concrete.

CONF Field: no History: no .fil: yes .dat: yes

Number of cracks at a concrete material point.

PEQC Field: yes History: yes .fil: yes .dat: yes

All equivalent plastic strains when the model has more than one yield/failure surface.

PEQC*n* Field: no History: yes .fil: no .dat: yes

*n*th equivalent plastic strain ($n = 1, 2, 3, 4$).

For jointed materials: PEQC provides equivalent plastic strains for all four possible systems (three joints - PEQC1, PEQC2, PEQC3, and bulk material - PEQC4). This identifier also provides a yes/no flag (1/0 on the output database) telling if each individual system is currently yielding or not (AC YIELD: "actively yielding"; that is, the plastic strain changed during the increment).

For cap plasticity: PEQC provides equivalent plastic strains for all three possible yield/failure surfaces (Drucker-Prager failure surface - PEQC1, cap surface - PEQC2, and transition surface - PEQC3) and the total volumetric inelastic strain (PEQC4). All identifiers also provide a yes/no flag (1/0 on the output database) telling whether the yield surface is currently active or not (AC YIELD: "actively yielding", that is, the plastic strain changed during the increment).

When PEQC is requested as output to the output database, the active yield flags for each component are named AC YIELD1, AC YIELD2, etc. and take the value 1 or 0.

RD Field: yes History: yes .fil: yes .dat: yes

Relative density for cap plasticity.

VVF Field: yes History: yes .fil: yes .dat: yes

Void volume fraction for cap plasticity.

Concrete Damaged Plasticity

DAMAGEC Field: yes History: yes .fil: no .dat: yes

Compressive damage variable, d_c.

DAMAGET Field: yes History: yes .fil: no .dat: yes

Tensile damage variable, d_t.

SDEG Field: yes History: yes .fil: no .dat: yes

Scalar stiffness degradation variable, *d*.

PEEQ Field: yes History: yes .fil: yes .dat: yes

Equivalent plastic strain in uniaxial compression, which is defined as $\int \dot{\tilde{\varepsilon}}_c^{pl} dt$. This identifier also provides a yes/no flag (1/0 on the output database) telling if the material is currently undergoing compressive failure or not (AC YIELD: "actively yielding"; that is, the plastic strain changed during the increment).

Superelastic Material Quantities

TE Field: yes History: yes .fil: no .dat: no

All transformation strain tensor components.

TE*ij* Field: no History: yes .fil: no .dat: no

ij-component of transformation strain ($i \leq j \leq 3$).

TEEQ Field: yes History: yes .fil: no .dat: no

Equivalent transformation strain.

TEVOL Field: yes History: yes .fil: no .dat: no

Volumetric transformation strain.

MVF Field: yes History: yes .fil: no .dat: no

Fraction of martensite.

SEQUT Field: yes History: yes .fil: no .dat: yes

Equivalent uniaxial tensile stress.

EEQUT Field: yes History: yes .fil: no .dat: yes

Equivalent uniaxial tensile total strain.

Rebar Quantities

RBFOR Field: yes History: yes .fil: yes .dat: yes

Force in rebar.

RBANG Field: yes History: yes .fil: yes .dat: yes

Angle in degrees between rebar and the user-specified isoparametric direction. Available only for shell, membrane, and surface elements.

RBROT Field: yes History: yes .fil: yes .dat: yes

Change in angle in degrees between rebar and the user-specified isoparametric direction. Available only for shell, membrane, and surface elements.

Heat Transfer Analysis

HFL Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the heat flux per unit area vector. The integration points for these values are located at the Gauss points.

HFLM Field: no History: yes .fil: no .dat: yes

Current magnitude of heat flux per unit area vector.

HFL*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the heat flux vector ($n = 1, 2, 3$).

GRADT Field: yes History: yes .fil: yes .dat: yes

Temperature gradient vector.

GRADT*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the temperature gradient vector ($n = 1, 2, 3$).

TEMPR Field: yes History: yes .fil: yes .dat: yes

Temperature rate.

Mass Diffusion Analysis

CONC Field: yes History: yes .fil: yes .dat: yes

Mass concentration.

ISOL Field: yes History: yes .fil: yes .dat: yes

Amount of solute at an integration point, calculated as the product of the mass concentration (CONC) and the integration point volume (IVOL).

MFL Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the concentration flux vector.

MFLM Field: no History: yes .fil: no .dat: yes

Current magnitude of the concentration flux vector.

MFL*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the concentration flux vector ($n = 1, 2, 3$).

Elements with Electrical Potential Degrees of Freedom

EPG Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the electrical potential gradient vector for a coupled thermal-electrical analysis, a coupled thermal-electrochemical analysis, or a fully coupled thermal-electrical-structural analysis. Current magnitude and components of the negative of the electrical potential gradient vector for a piezoelectric analysis.

EPGM Field: no History: yes .fil: no .dat: yes

Current magnitude of the electrical potential gradient vector.

EPG*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the electrical potential gradient vector for a coupled thermal-electrical analysis, a coupled thermal-electrochemical analysis, or a fully coupled thermal-electrical-structural analysis. Component *n* of the negative of the electrical potential gradient vector for a piezoelectric analysis. ($n = 1, 2, 3$) .

Piezoelectric Analysis

EFLX Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the electrical flux vector.

EFLXM Field: no History: yes .fil: no .dat: yes

Current magnitude of the electrical flux vector.

EFLX*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the electrical flux vector ($n = 1, 2, 3$).

Coupled Thermal-Electrical Elements

ECD Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the electrical current density.

ECDM Field: no History: yes .fil: no .dat: yes

Current magnitude of the electrical current density.

ECD*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the electrical current density vector ($n = 1, 2, 3$).

Coupled Thermal-Electrochemical Elements

ECDE Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the electrical current density in the fluid.

MFLE Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the ion flux vector in the electrolyte.

MFLS Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the species flux vector in the electrode.

CONCE Field: yes History: yes .fil: yes .dat: yes

Ion concentration in the fluid.

CONCS Field: yes History: yes .fil: yes .dat: yes

Species concentration in the solid electrode.

ELECPOTE Field: yes History: yes .fil: yes .dat: yes

Electric potential in the fluid.

ELECPOT Field: yes History: yes .fil: yes .dat: yes

Electric potential in the solid.

EPGE Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the fluid electrical potential gradient vector for a coupled thermal-electrochemical analysis.

CONCGE Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the ion concentration gradient vector for a coupled thermal-electrochemical analysis.

FLUIDVF Field: yes History: yes .fil: no .dat: no

Fluid volume fraction.

SOLIDVF Field: yes History: yes .fil: no .dat: no

Solid volume fraction.

BINDERVF Field: yes History: yes .fil: no .dat: no

Binder volume fraction.

ECHEMQ1 Field: yes History: yes .fil: no .dat: no

Heat generation volumetric flux of Butler-Volmer electric current, Q_1.

ECHEMQ2 Field: yes History: yes .fil: no .dat: no

Entropic heat generation volumetric flux, Q_2.

ECHEMQ3 Field: yes History: yes .fil: no .dat: no

Heat generation volumetric flux of electronic electric current, Q_3.

ECHEMQ4 Field: yes History: yes .fil: no .dat: no

Heat generation volumetric flux of ionic electric current, Q_4.

ECHEMQ5 Field: yes History: yes .fil: no .dat: no

Heat generation of volumetric ionic diffusive flux, Q_5.

ECHEMQ Field: yes History: yes .fil: no .dat: no

Sum of available heat generation terms, from Q_1 to Q_5 above.

CSURFAVG Field: yes History: yes .fil: no .dat: no

Average normalized surface concentration in the particles on the microscale.

ECDBV Field: yes History: yes .fil: no .dat: no

Butler-Volmer electrical current density.

LOGGRAINJG Field: yes History: yes .fil: no .dat: no

Logarithm of solid grains Jacobian as a measure of volumetric swelling.

Coupled Thermal-Electrochemical Elements, Particle Type–Related Variables

The following output variables are available only at electrodes. The subscript "i" in the output variable labels refers to the particle type. For example, if an electrode has three types of particles (i=1, 2, and 3), output variables CSURF_1, CSURF_2, and CSURF_3; OCP_1, OCP_2, and OCP_3; etc., are available.

The concentration below refers to that of lithium metal in the particles and is measured in moles per unit reference volume of the particle. While the output variables presented here are in terms of lithium metal, in general, you can replace lithium with another chemical species.

CSURF_*i* Field: yes History: yes .fil: no .dat: no

Absolute concentration of lithium metal on the surface of particle i ($1 \leq i \leq 3$).

OCP_*i* Field: yes History: yes .fil: no .dat: no

Open circuit potential in particle i ($1 \leq i \leq 3$).

CAVG_*i* Field: yes History: yes .fil: no .dat: no

Absolute average concentration of lithium metal in particle i ($1 \leq i \leq 3$) for the swelling computations.

TOTALC_*i* Field: yes History: yes .fil: no .dat: no

Total amount of lithium metal in particle i ($1 \leq i \leq 3$).

ECDBV_*i* Field: yes History: yes .fil: no .dat: no

Butler-Volmer current density in particle i ($1 \leq i \leq 3$). The value is output as current per unit volume, computed as $a_s I_{BV}$ for particle i. For more details, refer to *Governing Equations*.

OVERPOT_*i* Field: yes History: yes .fil: no .dat: no

Overpotential in particle i ($1 \leq i \leq 3$).

ECHEMQ1_*i* Field: yes History: yes .fil: no .dat: no

Ohmic loss at the solid-liquid interface for particle i ($1 \leq i \leq 3$).

ECHEMQ2_*i* Field: yes History: yes .fil: no .dat: no

Entropy generation in particle i ($1 \leq i \leq 3$).

SOC_*i* Field: yes History: yes .fil: no .dat: no

State of charge in particle i ($1 \leq i \leq 3$).

VOLE_*i* Field: yes History: yes .fil: no .dat: no

Volumetric swelling (Jacobian) in particle i ($1 \leq i \leq 3$).

Cohesive Elements

MAXSCRT Field: yes History: yes .fil: no .dat: yes

Maximum nominal stress damage initiation criterion.

MAXECRT Field: yes History: yes .fil: no .dat: yes

Maximum nominal strain damage initiation criterion.

QUADSCRT Field: yes History: yes .fil: no .dat: yes

Quadratic nominal stress damage initiation criterion.

QUADECRT Field: yes History: yes .fil: no .dat: yes

Quadratic nominal strain damage initiation criterion.

DMICRT Field: yes History: yes .fil: yes .dat: yes

All active components of the damage initiation criteria.

SDEG Field: yes History: yes .fil: yes .dat: yes

Overall scalar stiffness degradation.

STATUS Field: yes History: yes .fil: yes .dat: yes

Status of the element (the status of an element is 1.0 if the element is active, 0.0 if the element is not).

The status output is added automatically by the analysis.

MMIXDME Field: yes History: yes .fil: no .dat: no

Mode mix ratio during damage evolution. It has a value of -1.0 before initiation of damage.

MMIXDMI Field: yes History: yes .fil: no .dat: no

Mode mix ratio at damage initiation. It has a value of -1.0 before initiation of damage.

Low-Cycle Fatigue Analysis

CYCLEINI Field: yes History: yes .fil: no .dat: no

Number of cycles to initialize the damage at the material point.

SDEG Field: yes History: yes .fil: yes .dat: yes

Overall scalar stiffness degradation.

STATUS Field: yes History: yes .fil: yes .dat: yes

Status of the element (the status of an element is 1.0 if the element is active, 0.0 if the element is not).

The status output is added automatically by the analysis.

Pore Pressure Analysis

VOIDR Field: yes History: yes .fil: yes .dat: yes

Void ratio.

POR Field: yes History: yes .fil: yes .dat: yes

Pore pressure.

SAT Field: yes History: yes .fil: yes .dat: yes

Saturation.

GELVR Field: yes History: yes .fil: yes .dat: yes

Gel volume ratio.

FLUVR Field: yes History: yes .fil: yes .dat: yes

Total fluid volume ratio.

FLVEL Field: yes History: yes .fil: yes .dat: yes

Current magnitude and components of the pore fluid effective velocity vector.

FLVELM Field: no History: yes .fil: no .dat: yes

Current magnitude of the pore fluid effective velocity vector.

FLVEL*n* Field: no History: yes .fil: no .dat: yes

Component n of the pore fluid effective velocity vector ($n = 1, 2, 3$).

SROCK Field: yes History: no .fil: no .dat: no

All active components of rock mechanics effective stress.

Pore Pressure Cohesive Elements

GFVR Field: yes History: yes .fil: yes .dat: yes

Gap flow volume rate.

PFOPEN Field: yes History: yes .fil: yes .dat: yes

Pore pressure fracture opening.

LEAKVRT Field: yes History: yes .fil: yes .dat: yes

Leak-off flow rate at the top of the element.

LEAKVRB Field: yes History: yes .fil: yes .dat: yes

Leak-off flow rate at the bottom of the element.

ALEAKVRT Field: yes History: yes .fil: yes .dat: yes

Accumulated leak-off volume at the top of the element.

ALEAKVRB Field: yes History: yes .fil: yes .dat: yes

Accumulated leak-off volume at the bottom of the element.

FLDVEL Field: yes History: yes .fil: yes .dat: yes

Material point fluid velocity.

Coupled Slurry Concentration-Pore Pressure Elements

SLURRYVF Field: yes History: yes .fil: no .dat: no

Volumetric concentration of proppant particles in the slurry (slurry concentration).

SLURRYAF Field: yes History: yes .fil: no .dat: no

Volume of proppant particles in the slurry per unit area.

THKFTCKT Field: yes History: yes .fil: no .dat: no

Filter cake thickness with leak-off at top surface.

THKFTCKB Field: yes History: yes .fil: no .dat: no

Filter cake thickness with leak-off at bottom surface.

Porous Metal Plasticity Quantities

RD
Field: yes History: yes .fil: yes .dat: yes
Relative density.

VVF
Field: yes History: yes .fil: yes .dat: yes
Void volume fraction.

VVFG
Field: yes History: yes .fil: yes .dat: yes
Void volume fraction due to void growth.

VVFN
Field: yes History: yes .fil: yes .dat: yes
Void volume fraction due to void nucleation.

Two-Layer Viscoplasticity Quantities

VS
Field: yes History: yes .fil: yes .dat: yes
Stress in the elastic-viscous network.

VS*ij*
Field: no History: yes .fil: no .dat: yes
ij-component of stress in the elastic-viscous network ($i \leq j \leq 3$).

PS
Field: yes History: yes .fil: yes .dat: yes
Stress in the elastic-plastic network.

PS*ij*
Field: no History: yes .fil: no .dat: yes
ij-component of stress in the elastic-plastic network ($i \leq j \leq 3$).

VE
Field: yes History: yes .fil: yes .dat: yes
Viscous strain in the elastic-viscous network.

VE*ij*
Field: no History: yes .fil: no .dat: yes
ij-component of viscous strain in the elastic-viscous network ($i \leq j \leq 3$).

PE
Field: yes History: yes .fil: yes .dat: yes
Plastic strain in the elastic-plastic network.

PE*ij*
Field: no History: yes .fil: no .dat: yes
ij-component of plastic strain in the elastic-plastic network ($i \leq j \leq 3$).

VEEQ
Field: yes History: yes .fil: no .dat: yes
Equivalent viscous strain in the elastic-viscous network, defined as $\int_0^t \dot{\bar{\varepsilon}}^v dt$.

PEEQ
Field: yes History: yes .fil: no .dat: yes
Equivalent plastic strain in the elastic-plastic network, defined as $\int_0^t \dot{\bar{\varepsilon}}^{pl} dt$.

Geometric Quantities

COORD Field: yes History: yes .fil: yes .dat: yes

Coordinates of the integration point in elements and rebar. These are the current coordinates if the large-displacement formulation is being used.

IVOL Field: yes History: yes .fil: yes .dat: yes

Integration point volume. Section point volume in the case of beams and shells. (Not available for eigenfrequency extraction, eigenvalue buckling prediction, complex eigenfrequency extraction, or linear dynamics procedures. Available only for continuum and structural elements not using general beam or shell section definitions.)

LOCALDIR*n* Field: automatic History: no .fil: no .dat: no

Direction cosines of the local material directions for an anisotropic hyperelastic material model. This variable is output automatically if any other element field output is requested for an anisotropic hyperelastic material (see *Output*).

Accuracy Indicators

SJP Field: no History: no .fil: yes .dat: yes

Strain jumps at nodes.

Random Response Analysis

The following variables (beginning with R) are available only for random response dynamic analysis:

RS Field: yes History: yes .fil: yes .dat: yes

Root mean square of all stress components.

RS*ij* Field: no History: yes .fil: no .dat: yes

Root mean square of ij-component of stress ($i \leq j \leq 3$).

RMISES Field: yes History: yes .fil: no .dat: no

Root mean square of Mises equivalent stress.

RE Field: yes History: yes .fil: yes .dat: yes

Root mean square of all strain components.

RE*ij* Field: no History: yes .fil: no .dat: yes

Root mean square of ij-component of strain ($i \leq j \leq 3$).

RCTF Field: no History: yes .fil: yes .dat: yes

RMS values of all components of connector total forces and moments.

RCTF*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector total force component *n* ($n = 1, 2, 3$).

RCTM*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector total moment component *n* ($n = 1, 2, 3$).

RCEF Field: no History: yes .fil: yes .dat: yes

RMS values of all components of connector elastic forces and moments.

RCEF*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector elastic force component *n* ($n = 1, 2, 3$).

RCEM*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector elastic moment component *n* ($n = 1, 2, 3$).

RCVF Field: no History: yes .fil: yes .dat: yes

RMS values of all components of connector viscous forces and moments.

RCVF*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector viscous force component *n* ($n = 1, 2, 3$).

RCVM*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector viscous moment component *n* ($n = 1, 2, 3$).

RCRF Field: no History: yes .fil: yes .dat: yes

RMS values of all components of connector reaction forces and moments.

RCRF*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector reaction force component *n* ($n = 1, 2, 3$).

RCRM*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector reaction moment component *n* ($n = 1, 2, 3$).

RCSF Field: no History: yes .fil: yes .dat: yes

RMS values of all components of connector friction forces and moments.

RCSF*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector friction force component *n* ($n = 1, 2, 3$).

RCSM*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector friction moment component *n* ($n = 1, 2, 3$).

RCSFC Field: no History: yes .fil: no .dat: yes

RMS value of connector friction force in the direction of the instantaneous slip direction. Available only if friction is defined in the slip direction.

RCU Field: no History: yes .fil: yes .dat: yes

RMS values of all components of connector relative displacements and rotations.

RCU*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector relative displacement in the *n*-direction ($n = 1, 2, 3$).

RCUR*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector relative rotation in the *n*-direction ($n = 1, 2, 3$).

RCCU Field: no History: yes .fil: yes .dat: yes

RMS values of all components of connector constitutive displacements and rotations.

RCCU*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector constitutive displacement in the *n*-direction ($n = 1, 2, 3$).

RCCUR*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector constitutive rotation in the *n*-direction ($n = 1, 2, 3$).

RCNF Field: no History: yes .fil: yes .dat: yes

RMS values of all components of connector friction-generating contact forces and moments.

RCNF*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector friction-generating contact force component *n* ($n = 1, 2, 3$).

RCNM*n* Field: no History: yes .fil: no .dat: yes

RMS value of connector friction-generating contact moment component *n* ($n = 1, 2, 3$).

RCNFC Field: no History: yes .fil: no .dat: yes

RMS values of connector friction-generating contact force components in the instantaneous slip direction. Available only if friction is defined in the slip direction.

Steady-State Dynamic Analysis

The following variables (beginning with P) are available only for steady-state (frequency domain) dynamic analysis. These variables include both the magnitude and phase angle for all components. Phase angles are given in degrees. In the data file there are two lines of output for each request. The first line contains the magnitude, and the second line (indicated by the SSD footnote) contains the phase angle. In the results file the magnitudes of all components are first, followed by the phase angles of all components.

PHS Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of all stress components.

PHS*ij* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of ij-component of stress ($i \leq j \leq 3$).

PHE Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of all strain components.

PHE*ij* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of ij-component of strain ($i \leq j \leq 3$).

PHEPG Field: no History: no .fil: yes .dat: yes

Magnitude and phase angles of the electrical potential gradient vector.

PHEPG*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of component *n* of the electrical potential gradient ($n = 1, 2, 3$).

PHEFL Field: no History: no .fil: yes .dat: yes

Magnitude and phase angles of the electrical flux vector.

PHEFL*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of component *n* of the electrical flux vector ($n = 1, 2, 3$).

PHMFL Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of mass flow rate. Available only for fluid link elements.

PHMFT Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of total mass flow. Available only for fluid link elements.

PHCTF Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector total forces and moments.

PHCTF*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector total force component *n* ($n = 1, 2, 3$).

PHCTM*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector total moment component *n* ($n = 1, 2, 3$).

PHCEF Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector elastic forces and moments.

PHCEF*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector elastic force component *n* ($n = 1, 2, 3$).

PHCEM*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector elastic moment component *n* ($n = 1, 2, 3$).

PHCVF Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector viscous forces and moments.

PHCVF*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector viscous force component *n* ($n = 1, 2, 3$).

PHCVM*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector viscous moment component *n* ($n = 1, 2, 3$).

PHCRF Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector reaction forces and moments.

PHCRF*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector reaction force component *n* ($n = 1, 2, 3$).

PHCRM*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector reaction moment component *n* ($n = 1, 2, 3$).

PHCSF
Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector friction forces and moments.

PHCSF*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector friction force component *n* ($n = 1, 2, 3$).

PHCSM*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector friction moment component *n* ($n = 1, 2, 3$).

PHCSFC
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector friction force in the direction of the instantaneous slip direction. Available only if friction is defined in the slip direction.

PHCU
Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector relative displacements and rotations.

PHCU*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector relative displacement in the *n*-direction ($n = 1, 2, 3$).

PHCUR*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector relative rotation in the *n*-direction ($n = 1, 2, 3$).

PHCCU
Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector constitutive displacements and rotations.

PHCCU*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector constitutive displacement in the *n*-direction ($n = 1, 2, 3$).

PHCCUR*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector constitutive rotation in the *n*-direction ($n = 1, 2, 3$).

PHCV
Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector relative velocities.

PHCV*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector relative velocity in the *n*-direction ($n = 1, 2, 3$).

PHCVR*n*
Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector relative angular velocity in the *n*-direction ($n = 1, 2, 3$).

PHCA
Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector relative accelerations.

PHCA*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector relative acceleration in the *n*-direction ($n = 1, 2, 3$).

PHCAR*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector relative angular acceleration in the *n*-direction ($n = 1, 2, 3$).

PHCNF Field: no History: no .fil: yes .dat: yes

Magnitude and phase of all components of connector friction-generating contact forces and moments.

PHCNF*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector friction-generating contact force component *n* ($n = 1, 2, 3$).

PHCNM*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector friction-generating contact moment component *n* ($n = 1, 2, 3$).

PHCNFC Field: no History: no .fil: no .dat: yes

Magnitude and phase of connector friction-generating contact force in the instantaneous slip direction. Available only if friction is defined in the slip direction.

PHCIVC Field: no History: no .fil: yes .dat: yes

Magnitude and phase of connector instantaneous velocity in the slip direction. Available only if friction is defined in the slip direction.

Failure with Progressive Damage

SDEG Field: yes History: yes .fil: no .dat: no

Scalar stiffness degradation variable.

DMICRT Field: yes History: yes .fil: no .dat: no

All active components of the damage initiation criteria.

DMICRTMAX Field: yes History: no .fil: no .dat: no

Maximum damage initiation criteria value among all section points, all damage initiation criteria, and all time increments up to the current time.

This output variable generates three output quantities as follows:

DMICRTMAXVAL outputs the maximum damage initiation value.

DMICRTPOS outputs the section point in the layer in which the maximum damage initiation value occurred. For solid elements, the output value is one.

DMICRTTYPE outputs a value that represents the damage initiation criteria type that reached the maximum value in the element as follows:

For elements that have failure with progressive damage: 1-DUCTCRT, 2-SHRCRT, 4-FLDCRT, 5-MSFLDCRT, and 6-FLSDCRT.

For elements that have fiber-reinforced material damage: 11-HSNFTCRT, 12-HSNFCCRT, 13-HSNMTCRT, and 14-HSNMCCRT.

For cohesive elements with traction-separation behavior: 21-MAXSCRT, 22-MAXECRT, 23-QUADSCRT, and 24-QUADECRT.

DUCTCRT Field: yes History: yes .fil: no .dat: no

Ductile damage initiation criterion.

HCCRT Field: no History: yes .fil: no .dat: yes

Hosford-Coulomb damage initiation criterion.

SHRCRT Field: yes History: yes .fil: no .dat: no

Shear damage initiation criterion.

FLDCRT Field: yes History: yes .fil: no .dat: no

Forming limit diagram (FLD) damage initiation criterion.

FLSDCRT Field: yes History: yes .fil: no .dat: no

Forming limit stress diagram (FLSD) damage initiation criterion.

MSFLDCRT Field: yes History: yes .fil: no .dat: no

Müschenborn-Sonne forming limit stress diagram (MSFLD) damage initiation criterion.

ERPRATIO Field: yes History: yes .fil: no .dat: no

Ratio of principal strain rates, α, used for the MSFLD damage initiation criterion.

SHRRATIO Field: yes History: yes .fil: no .dat: no

Shear stress ratio, $\theta_s = (q + k_s p) / \tau_{\max}$, used for the shear damage initiation criterion.

General Strain/Stress-Based Damage

DMICRT Field: yes History: yes .fil: no .dat: no

All active components of the damage initiation criteria.

MSTRESSCRT Field: yes History: yes .fil: no .dat: no

Maximum stress-based damage initiation criterion.

MSTRAINCRT Field: yes History: yes .fil: no .dat: no

Maximum strain-based damage initiation criterion.

TSAIWUCRT Field: yes History: yes .fil: no .dat: no

Tsai-Wu stress-based damage initiation criterion.

TSAIWUECRT Field: yes History: yes .fil: no .dat: no

Tsai-Wu strain-based damage initiation criterion.

DMIFI Field: yes History: yes .fil: no .dat: no

All active components of the damage initiation failure indices.

MSTRESSFI Field: yes History: yes .fil: no .dat: no

Maximum stress-based damage initiation failure index.

MSTRAINFI Field: yes History: yes .fil: no .dat: no

Maximum strain-based damage initiation failure index.

TSAIWUFI Field: yes History: yes .fil: no .dat: no

Tsai-Wu stress-based damage initiation failure index.

TSAIWUEFI Field: yes History: yes .fil: no .dat: no

Tsai-Wu strain-based damage initiation failure index.

Fiber-Reinforced Materials Damage

HSNFTCRT Field: yes History: yes .fil: no .dat: yes

Hashin fiber tensile damage initiation criterion.

HSNFCCRT Field: yes History: yes .fil: no .dat: yes

Hashin fiber compressive damage initiation criterion.

HSNMTCRT Field: yes History: yes .fil: no .dat: yes

Hashin matrix tensile damage initiation criterion.

TSINVMTCRT Field: yes History: yes .fil: no .dat: yes

Transversely isotropic stress invariant-based matrix tensile damage initiation criterion.

TSINVMCCRT Field: yes History: yes .fil: no .dat: yes

Transversely isotropic stress invariant-based matrix compression damage initiation criterion.

HSNMCCRT Field: yes History: yes .fil: no .dat: yes

Hashin matrix compressive damage initiation criterion.

LARCMCCRT Field: yes History: yes .fil: no .dat: yes

LaRC05 matrix cracking damage initiation criterion.

LARCFKCRT Field: yes History: yes .fil: no .dat: yes

LaRC05 fiber kinking damage initiation criterion.

LARCFSCRT Field: yes History: yes .fil: no .dat: yes

LaRC05 fiber splitting damage initiation criterion.

LARCFTCRT Field: yes History: yes .fil: no .dat: yes

LaRC05 fiber tension damage initiation criterion.

DMICRT Field: yes History: yes .fil: yes .dat: yes

All active components of the damage initiation criteria.

DAMAGEFT Field: yes History: yes .fil: yes .dat: yes

Fiber tensile damage variable.

DAMAGEFC Field: yes History: yes .fil: yes .dat: yes

Fiber compressive damage variable.

DAMAGEMT Field: yes History: yes .fil: yes .dat: yes

Matrix tensile damage variable.

DAMAGEMC Field: yes History: yes .fil: yes .dat: yes

Matrix compressive damage variable.

DAMAGESHR Field: yes History: yes .fil: yes .dat: yes

Shear damage variable.

STATUS Field: yes History: yes .fil: yes .dat: yes

Status of the element (the status of an element is 1.0 if the element is active, 0.0 if the element is not).

The status output is added automatically by the analysis.

Structural Relaxation Quantities

TFICT Field: yes History: yes .fil: no .dat: no

Fictive temperature.

Cure Modeling Quantities

CURESE Field: yes History: yes .fil: no .dat: no

All cure shrinkage strain components.

CURESE*ij* Field: yes History: yes .fil: no .dat: no

ij-component of cure shrinkage strain ($i \leq j \leq 3$).

DDOCRDTEMP Field: yes History: yes .fil: no .dat: no

Derivative of the degree of cure rate with respect to temperature, $\frac{\partial \dot{\alpha}}{\partial T}$.

DOC Field: yes History: yes .fil: no .dat: no

Degree of cure, α.

DOCR Field: yes History: yes .fil: no .dat: no

Degree of cure rate, $\dot{\alpha}$.

TG Field: yes History: yes .fil: no .dat: no

Glass transition temperature.

TGTDIFF Field: yes History: yes .fil: no .dat: no

The difference between the glass transition temperature and the temperature, $\theta_g - \theta$.

Plasticity Corrections Quantities

GKEEQ Field: yes History: yes .fil: no .dat: no

Glinka equivalent strain.

GKPEEQ Field: yes History: yes .fil: no .dat: no

Glinka equivalent plastic strain.

GKSEQ Field: yes History: yes .fil: no .dat: no

Glinka equivalent stress.

NBEEQ Field: yes History: yes .fil: no .dat: no

Neuber equivalent strain.

NBPEEQ Field: yes History: yes .fil: no .dat: no

Neuber equivalent plastic strain.

NBSEQ Field: yes History: yes .fil: no .dat: no

Neuber equivalent stress.

Element Centroidal Variables

The output variables listed below are available in Abaqus/Standard.

EMB Field: yes History: yes .fil: no .dat: no

All components of the magnetic flux density vector.

EMH Field: yes History: yes .fil: no .dat: no

All components of the magnetic field vector.

EME Field: yes History: yes .fil: no .dat: no

All components of the electric field vector.

EMCD Field: yes History: yes .fil: no .dat: no

All components of the eddy current density vector in conducting regions.

EMCDA Field: yes History: yes .fil: no .dat: no

Magnitude and components of the applied volume current density vector.

EMJH Field: yes History: yes .fil: no .dat: no

Rate of Joule heat dissipation (amount of heat dissipated per unit volume per unit time) in conductor regions.

EMBF Field: yes History: yes .fil: no .dat: no

Magnetic body force intensity (force per unit volume) vector due to induced current in conductor regions.

EMBFC Field: yes History: yes .fil: no .dat: no

Complex magnetic body force intensity (force per unit volume) vector in conductor regions in a time-harmonic eddy current analysis.

References:

- *Eddy Current Analysis*
- *Magnetostatic Analysis*

Element Section Variables

The output variables listed below are available in Abaqus/Standard.

SF Field: yes History: yes .fil: yes .dat: yes

All section force and moment components.

SF*n* Field: no History: yes .fil: no .dat: yes

Section force per unit width of component *n* ($n = 1, 2, 3, 4, 5$ for conventional shells; $n = 1, 2, 3, 4, 5, 6$ for continuum shells; $n = 1, 2, 3$ for beams).

SM*n* Field: no History: yes .fil: no .dat: yes

Section moment per unit width of component *n* ($n = 1, 2, 3$).

SORIENT Field: yes History: no .fil: no .dat: no

Composite shell section orientations.

BIMOM Field: yes History: yes .fil: no .dat: yes

Bimoment of beam cross-section. Available only for open-section beam elements.

ESF1 Field: yes History: yes .fil: yes .dat: yes

Effective axial force for beams and pipes subjected to pressure loading. Available for all stress/displacement procedure types except response spectrum and random response.

SSAVG Field: yes History: no .fil: yes .dat: yes

All average shell section stress components.

SSAVG*n* Field: no History: yes .fil: no .dat: yes

Average shell section stress component *n* ($n = 1, 2, 3, 4, 5, 6$).

SE Field: yes History: yes .fil: yes .dat: yes

All section strain, curvature change, and twist components.

SE*n* Field: no History: yes .fil: no .dat: yes

Section strain component *n* ($n = 1, 2, 3, 4, 5, 6$ for shells; $n = 1, 2, 3$ for beams).

SK Field: yes History: yes .fil: yes .dat: yes

Section curvature change and twist components.

SK*n* Field: no History: yes .fil: no .dat: yes

Section curvature change or twist *n* ($n = 1, 2, 3$).

BICURV Field: yes History: yes .fil: no .dat: yes

Bicurvature of beam cross-section. Available only for open-section beam elements.

MAXSS Field: no History: no .fil: yes .dat: yes

Maximum axial stress on the section. (This variable can be used with the following types of general beam section definitions: standard library cross-sections, linear generalized cross-sections, or meshed cross-sections with specified output section points. If the output section points are specified, the MAXSS output will be the maximum of the stresses at the user-specified points.)

STH Field: yes History: yes .fil: yes .dat: yes

Section thickness (current thickness for SAX1, SAX2, SAX2T, S3/S3R, S4, S4R, SAXA1*N*, SAXA2*N*, and all membrane elements if the large-displacement formulation is used; initial thickness for all other cases).

SVOL Field: yes History: yes .fil: yes .dat: yes

Integrated section volume. (Not available for eigenfrequency extraction, eigenvalue buckling prediction, complex eigenfrequency extraction, or linear dynamics procedures. Available only for continuum and structural elements not using general beam or shell section definitions.)

SPE Field: yes History: yes .fil: yes .dat: yes

All generalized plastic strain components. Available only for inelastic nonlinear response in a general beam section.

SPE*n* Field: no History: yes .fil: no .dat: yes

Generalized plastic strain component n ($n = 1, 2, 3, 4$). Representing axial plastic strain, curvature change about the local 1-axis, curvature change about the local 2-axis, and twist of the beam. Available only for inelastic nonlinear response in a general beam section.

SEPE Field: yes History: yes .fil: yes .dat: yes

All equivalent plastic strains. Available only for inelastic nonlinear response in a general beam section.

SEPE*n* Field: no History: yes .fil: no .dat: yes

Equivalent plastic strain component n ($n = 1, 2, 3, 4$). Representing axial plastic strain, curvature change about the local 1-axis, curvature change about the local 2-axis, and twist of the beam. Available only for inelastic nonlinear response in a general beam section.

UVARPT Field: yes History: yes .fil: no .dat: no

Element user-defined output variables.

UVARPT*n* Field: yes History: yes .fil: no .dat: no

Element user-defined output variable n.

References:

- *Element Output*
- *Writing Element Output to the Output Database*

Frame Elements

SEE Field: yes History: yes .fil: yes .dat: yes

All elastic section axial, curvature, and twist strain components.

SEE1 Field: no History: yes .fil: no .dat: yes

Elastic axial strain component.

SKE*n* Field: no History: yes .fil: no .dat: yes

Elastic section curvature or twist strain component ($n = 1, 2, 3$).

SEP Field: yes History: yes .fil: yes .dat: yes

All plastic axial displacements and rotations at the element's ends. This identifier also provides a yes/no flag telling if the frame element's end section is currently yielding or not (AC YIELD: “actively yielding”; that is, the plastic strain changed during the increment) and a yes/no/na flag telling if buckling occurred in the strut response (AC BUCKL) or is not applicable. AC YIELD and AC BUCKL are not available in the output database.

SEP1 Field: no History: yes .fil: no .dat: yes

Plastic axial displacement at the element's ends.

SKP*n* Field: no History: yes .fil: no .dat: yes

Plastic rotations, either bending or twisting, at the element's ends ($n = 1, 2, 3$).

SALPHA Field: yes History: yes .fil: yes .dat: yes

All generalized backstress components at the element's ends.

SALPHA*n* Field: no History: yes .fil: no .dat: yes

Generalized backstress at the element's ends ($n = 1, 2, 3, 4$). The first component is the axial section backstress, followed by two bending backstress components and the twist backstress component.

Whole Element Variables

The output variables listed below are available in Abaqus/Standard.

LOADS Field: no History: no .fil: yes .dat: yes

Current values of distributed loads (not available for nonuniform loads).

FOUND Field: no History: no .fil: yes .dat: yes

Current values of foundation pressures.

FLUXS Field: yes History: no .fil: yes .dat: yes

Current values of distributed (heat or concentration) fluxes (not available for nonuniform fluxes), including those imported using the HFL co-simulation field ID.

CHRGS Field: no History: no .fil: yes .dat: yes

Current values of distributed electrical charges.

ECURS Field: no History: no .fil: yes .dat: yes

Current values of distributed electrical currents.

ELEN Field: yes History: yes .fil: yes .dat: yes

All energy magnitudes in the element. A limited number of them are available for direct-solution steady-state dynamic and subspace-based steady-state dynamic analyses. Mode-based steady-state dynamic analyses support computation of kinetic and strain energies as well as the energy loss due to viscous and structural damping (material and global).

ELKE Field: yes History: yes .fil: no .dat: yes

Total kinetic energy in the element. In steady-state dynamic and frequency extraction analyses, this is the cyclic mean value. In frequency extraction analyses, the value of total kinetic energy in the element is normalized. Normalization is performed for each eigenmode separately, such that the kinetic and strain energies for the whole model add up to one.

ELKEA Field: yes History: no .fil: no .dat: no

Total kinetic energy amplitude in the element. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ELKEP Field: yes History: no .fil: no .dat: no

Total kinetic energy peak value in the element. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ELSE Field: yes History: yes .fil: no .dat: yes

Total elastic strain energy in the element. When the Mullins effect is modeled with hyperelastic materials, this quantity represents only the recoverable part of energy in the element. In steady-state dynamic and frequency extraction analyses, this is the cyclic mean value. In frequency extraction analyses, the value of total elastic strain energy in the element is normalized. Normalization is performed for each eigenmode separately, such that the kinetic and strain energies for the whole model add up to one.

ELSEA Field: yes History: no .fil: no .dat: no

Total elastic strain energy amplitude in the element. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ELSEP Field: yes History: no .fil: no .dat: no

Total elastic strain energy peak value in the element. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ELPD Field: yes History: yes .fil: no .dat: yes

Total energy dissipated in the element by rate-independent and rate-dependent plastic deformation. For superelastic materials, this variable also includes recoverable phase-transformation energy. This output variable is not available for steady-state dynamic analysis.

ELCD Field: yes History: yes .fil: no .dat: yes

Total energy dissipated in the element by creep, swelling, viscoelasticity, and energy associated with viscous regularization for cohesive elements. Not available for steady-state dynamic analysis.

ELVD Field: yes History: yes .fil: no .dat: yes

Total energy dissipated in the element by viscous effects, not including energy dissipated by static stabilization or viscoelasticity. In mode-based and direct-solution steady-state dynamic analyses, only field output in the output database is supported for this variable.

ELVDE Field: yes History: no .fil: no .dat: no

Total energy dissipated in the element by viscous effects due to the material damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ELVDG Field: yes History: no .fil: no .dat: no

Total energy dissipated in the element by viscous effects due to the global damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ELHD Field: yes History: no .fil: no .dat: no

Total energy dissipated in the element due to structural damping. This variable includes energy loss due to the material and global structural damping and is available only in mode-based and direct-solution steady-state dynamic analyses.

ELHDE Field: yes History: no .fil: no .dat: no

Total energy dissipated in the element due to the material structural damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ELHDG Field: yes History: no .fil: no .dat: no

Total energy dissipated in the element due to the global structural damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ELSD Field: yes History: yes .fil: no .dat: yes

Total energy dissipated in the element resulting from automatic static stabilization. Not available for steady-state dynamic analysis.

ELCTE Field: yes History: yes .fil: no .dat: yes

Total electrostatic energy in the element. Not available for steady-state dynamic analysis.

ELJD Field: yes History: yes .fil: no .dat: yes

Total electrical energy dissipated due to flow of current. Not available for steady-state dynamic analysis.

ELASE Field: yes History: yes .fil: no .dat: yes

Total “artificial” strain energy in the element (energy associated with constraints used to remove singular modes, such as hourglass control, and with constraints used to make the drill rotation follow the in-plane rotation of the shell element). Not available for steady-state dynamic analysis.

ELDMD Field: yes History: yes .fil: no .dat: yes

Total energy dissipated in the element by damage. Not available for steady-state dynamic analysis.

EHUMDFLUX Field: yes History: yes .fil: no .dat: no

Element internal heat energy due to nonuniform moving flux prescribed using user subroutine *UMDFLUX*. Available only in a pure heat transfer analysis.

EHUMDFLUXDEN Field: yes History: yes .fil: no .dat: no

Element internal heat energy density due to nonuniform moving flux prescribed using user subroutine *UMDFLUX*. Available only in a pure heat transfer analysis.

NFORC Field: yes History: yes .fil: yes .dat: yes

In all analysis procedures except mode-based and direct-solution steady-state dynamic analyses: forces at the nodes of an element from both the hourglass and the regular deformation modes of that element that represent the action of the element to the rest of the finite element model (internal forces at the element nodes in the global coordinate system).

In mode-based and direct-solution steady-state dynamic analyses: internal forces at the element nodes that represent the action of the element to the rest of the finite element model, including contributions from the element inertia, damping, and deformation (internal forces at the element nodes in the global coordinate system).

The specified position in data and results file requests is ignored.

NFORCSO Field: yes History: yes .fil: no .dat: no

Forces (NFORCSOn, with n=1, 2, 3) and moments (n=4, 5, 6) at the nodes of a beam element caused by the stress resultants in the element (internal forces in the beam section orientation coordinate system).

For beams in space force components correspond to the axial force (n=1) and transverse shear forces in the local 2-direction (n=2) and local 1-direction (n=3), respectively. For beams in a plane force components correspond to the axial force (n=1) and transverse shear forces in the local 2-direction (n=2), respectively.

For beams in space moment components correspond to bending about the local 1-axis (n=4) and local 2-axis (n=5) and twist about the beam axis (n=6), respectively. For beams in a plane moment component NFORCSO6 corresponds to bending about the local 1-axis.

This variable is not supported in mode-based and direct-solution steady-state dynamic analysis procedures.

GRAV Field: yes History: no .fil: no .dat: no

Uniformly distributed gravity load (measured as g, where g is the gravitational acceleration).

BF Field: yes History: no .fil: no .dat: no

Uniformly distributed body force.

CORIOMAG Field: yes History: no .fil: no .dat: no

Magnitude of Coriolis load.

ROTAMAG Field: yes History: no .fil: no .dat: no

Magnitude of rotary acceleration load.

CENTMAG Field: yes History: no .fil: no .dat: no

Magnitude of centrifugal load (measured as $\rho\omega^2$, where ρ is the mass density per unit volume and ω is the angular velocity).

CENTRIFMAG Field: yes History: no .fil: no .dat: no

Magnitude of centrifugal load (measured as ω^2, where ω is the angular velocity).

HBF Field: yes History: no .fil: no .dat: no

Heat body flux.

NFLUX Field: yes History: yes .fil: yes .dat: yes

Fluxes at the nodes of the element caused by the heat conduction or mass diffusion in the element (internal fluxes). (The specified position for data and output database file requests is ignored.)

NFL*n* Field: no History: yes .fil: no .dat: yes

Flux *n* at the nodes of the element ($n = 11, 12, \ldots$) caused by the heat conduction or mass diffusion in the element (internal fluxes). (The specified position for data and output database file requests is ignored.)

NCURS Field: yes History: yes .fil: yes .dat: yes

Electrical current at the nodes due to electrical conduction in the element.

FILM Field: no History: no .fil: yes .dat: yes

Current values of film conditions (not available for nonuniform films).

RAD Field: no History: no .fil: yes .dat: yes

Current values of radiation conditions.

EACTIVE Field: yes History: yes .fil: no .dat: no

Volume fraction of the material in the current element.

EVOL Field: yes History: yes .fil: yes .dat: yes

Current element volume. (Not available for eigenfrequency extraction, eigenvalue buckling prediction, complex eigenfrequency extraction, or linear dynamics procedures. Available only for continuum and structural elements not using general beam or shell section definitions.)

ESOL Field: yes History: yes .fil: yes .dat: yes

Amount of solute in an element, calculated as the sum of ISOL (amount of solute at an integration point) over all the integration points in the element.

ESDV Field: yes History: yes .fil: no .dat: no

Element solution-dependent variables.

ESDV*n* Field: yes History: yes .fil: no .dat: no

Element solution-dependent variable *n*.

UVARE Field: yes History: yes .fil: no .dat: no

Element user-defined output variables.

UVARE*n* Field: yes History: yes .fil: no .dat: no

Element user-defined output variable *n*.

BRADIUS Field: yes History: no .fil: no .dat: no

Original beam radius for a solid circular beam.

BLADEINTERFER Field: yes History: yes .fil: no .dat: no

Powder blade interference with the printed part (0-No interference, 1-Possible interference, and 2-Interference detected between the powder blade and the printed part). Available only for the built-in thermomechanical additive manufacturing process simulations.

References:

- *Element Output*
- *Writing Element Output to the Output Database*

Enriched Elements

STATUSXFEM Field: yes History: yes .fil: no .dat: no

Status of the enriched element. (The status of an enriched element is 1.0 if the element is completely cracked; 0.0 if the element is not. If the element is partially cracked, the value lies between 1.0 and 0.0.)

LOADSXFEM Field: yes History: yes .fil: no .dat: no

Distributed pressure loads applied to the XFEM-based crack surface.

Enriched Elements When the XFEM-Based LEFM Approach Is Used

ENRRTXFEM Field: yes History: yes .fil: no .dat: no

All components of strain energy release rate.

Enriched Elements in Fatigue Crack Growth Analysis

CYCLEINIXFEM Field: yes History: yes .fil: no .dat: no

Minimum number of cycles needed to satisfy the condition for the onset of fatigue crack growth at the enriched element.

CYCLEXFEM Field: yes History: yes .fil: no .dat: no

Number of cycles to fracture at the enriched element.

Enriched Elements with Pore Pressure Degrees of Freedom

GFVRXFEM
Field: yes History: yes .fil: no .dat: no

Gap fluid volume rate of the enriched element.

CRDCUTXFEM
Field: yes History: yes .fil: no .dat: no

Crack midpoint coordinates at the element edges of the enriched element.

PFOPENXFEM
Field: yes History: yes .fil: no .dat: no

Fracture opening of the enriched element.

PFOPENXFEMCOMP
Field: yes History: yes .fil: no .dat: no

Fracture opening at the element edges of the enriched element.

PORPRES
Field: yes History: yes .fil: no .dat: no

Fluid pressure of the enriched element.

PORPRESCOMP
Field: yes History: yes .fil: no .dat: no

Fluid pressure at the element edges of the enriched element.

LEAKVRTXFEM
Field: yes History: yes .fil: no .dat: no

Leak-off flow rate at the top cracked surface of the enriched element.

LEAKVRBXFEM
Field: yes History: yes .fil: no .dat: no

Leak-off flow rate at the bottom cracked surface of the enriched element.

ALEAKVRTXFEM
Field: yes History: yes .fil: no .dat: no

Accumulated leak-off flow volume per unit area at the top cracked surface of the enriched element.

ALEAKVRBXFEM
Field: yes History: yes .fil: no .dat: no

Accumulated leak-off flow volume per unit area at the bottom cracked surface of the enriched element.

Connector Elements

CTF
Field: yes History: yes .fil: yes .dat: yes

All components of connector total forces and moments.

CTF*n*
Field: no History: yes .fil: no .dat: yes

Connector total force component *n* ($n = 1, 2, 3$).

CTM*n*
Field: no History: yes .fil: no .dat: yes

Connector total moment component *n* ($n = 1, 2, 3$).

CEF
Field: yes History: yes .fil: yes .dat: yes

All components of connector elastic forces and moments.

CEF*n* Field: no History: yes .fil: no .dat: yes

Connector elastic force component *n* ($n = 1, 2, 3$).

CEM*n* Field: no History: yes .fil: no .dat: yes

Connector elastic moment component *n* ($n = 1, 2, 3$).

CUE Field: yes History: yes .fil: yes .dat: yes

Elastic displacements and rotations in all directions.

CUE*n* Field: no History: yes .fil: no .dat: yes

Elastic displacement in the *n*-direction ($n = 1, 2, 3$).

CURE*n* Field: no History: yes .fil: no .dat: yes

Elastic rotation in the *n*-direction ($n = 1, 2, 3$).

CUP Field: yes History: yes .fil: yes .dat: yes

Plastic relative displacements and rotations in all directions.

CUP*n* Field: no History: yes .fil: no .dat: yes

Plastic relative displacement in the *n*-direction ($n = 1, 2, 3$).

CURP*n* Field: no History: yes .fil: no .dat: yes

Plastic relative rotation in the *n*-direction ($n = 1, 2, 3$).

CUPEQ Field: no History: yes .fil: yes .dat: yes

Equivalent plastic relative displacements and rotations in all directions.

CUPEQ*n* Field: no History: yes .fil: no .dat: yes

Equivalent plastic relative displacement in the *n*-direction ($n = 1, 2, 3$).

CURPEQ*n* Field: no History: yes .fil: no .dat: yes

Equivalent plastic relative rotation in the *n*-direction ($n = 1, 2, 3$).

CUPEQC Field: no History: yes .fil: no .dat: yes

Equivalent plastic relative motion for a coupled plasticity definition.

CALPHAF Field: no History: yes .fil: yes .dat: yes

All components of connector kinematic hardening shift forces and moments.

CALPHAF*n* Field: no History: yes .fil: no .dat: yes

Connector kinematic hardening shift force component *n* ($n = 1, 2, 3$).

CALPHAM*n* Field: no History: yes .fil: no .dat: yes

Connector kinematic hardening shift moment component *n* ($n = 1, 2, 3$).

CVF Field: no History: yes .fil: yes .dat: yes

All components of connector viscous forces and moments.

CVF*n* Field: no History: yes .fil: no .dat: yes

Connector viscous force component *n* ($n = 1, 2, 3$).

CVM*n* Field: no History: yes .fil: no .dat: yes

Connector viscous moment component *n* ($n = 1, 2, 3$).

CSF Field: no History: yes .fil: yes .dat: yes

All components of connector friction forces and moments.

CSF*n* Field: no History: yes .fil: no .dat: yes

Connector friction force component *n* ($n = 1, 2, 3$).

CSM*n* Field: no History: yes .fil: no .dat: yes

Connector friction moment component *n* ($n = 1, 2, 3$).

CSFC Field: no History: yes .fil: no .dat: yes

Connector friction force in the instantaneous slip direction. Available only if friction is defined in the slip direction.

CNF Field: no History: yes .fil: yes .dat: yes

All components of connector friction-generating contact forces and moments.

CNF*n* Field: no History: yes .fil: no .dat: yes

Connector friction-generating contact force component *n* ($n = 1, 2, 3$).

CNM*n* Field: no History: yes .fil: no .dat: yes

Connector friction-generating contact moment component *n* ($n = 1, 2, 3$).

CNFC Field: no History: yes .fil: no .dat: yes

Connector friction-generating contact force in the instantaneous slip direction. Available only if friction is defined in the slip direction.

CDMG Field: no History: yes .fil: yes .dat: yes

All components of the overall damage variable.

CDMG*n* Field: no History: yes .fil: no .dat: yes

Overall damage variable component *n* ($n = 1, 2, 3$).

CDMGR*n* Field: no History: yes .fil: no .dat: yes

Overall damage variable component *n* ($n = 1, 2, 3$).

CDIF Field: no History: yes .fil: yes .dat: yes

Components of connector force-based damage initiation criterion in all directions.

CDIF*n* Field: no History: yes .fil: no .dat: yes

Connector force-based damage initiation criterion in the *n*-translation direction ($n = 1, 2, 3$).

CDIFR*n* Field: no History: yes .fil: no .dat: yes

Connector force-based damage initiation criterion in the *n*-rotation direction ($n = 1, 2, 3$).

CDIFC Field: no History: yes .fil: no .dat: yes

Connector force-based damage initiation criterion in the instantaneous slip direction.

CDIM Field: no History: yes .fil: yes .dat: yes

Components of connector motion-based damage initiation criterion in all directions.

CDIM*n* Field: no History: yes .fil: no .dat: yes

Connector motion-based damage initiation criterion in the *n*-translation direction ($n = 1, 2, 3$).

CDIMR*n* Field: no History: yes .fil: no .dat: yes

Connector motion-based damage initiation criterion in the *n*-rotation direction ($n = 1, 2, 3$).

CDIMC Field: no History: yes .fil: no .dat: yes

Connector motion-based damage initiation criterion in the instantaneous slip direction.

CDIP Field: no History: yes .fil: yes .dat: yes

Components of connector plastic motion-based damage initiation criterion in all directions.

CDIP*n* Field: no History: yes .fil: no .dat: yes

Connector plastic motion-based damage initiation criterion in the *n*-translation direction ($n = 1, 2, 3$).

CDIPR*n* Field: no History: yes .fil: no .dat: yes

Connector plastic motion-based damage initiation criterion in the *n*-rotation direction ($n = 1, 2, 3$).

CDIPC Field: no History: yes .fil: no .dat: yes

Connector plastic motion-based damage initiation criterion in the instantaneous slip direction.

CSLST Field: no History: yes .fil: yes .dat: yes

All flags for connector stop and connector lock status.

CSLST*i* Field: no History: yes .fil: no .dat: yes

Flag for connector stop and connector lock status in the *i*-direction ($i = 1, \ldots, 6$).

CASU Field: no History: yes .fil: yes .dat: yes

Components of accumulated slip in all directions.

CASU*n* Field: no History: yes .fil: no .dat: yes

Connector accumulated slip in the *n*-direction ($n = 1, 2, 3$).

CASUR*n* Field: no History: yes .fil: no .dat: yes

Connector angular accumulated slip in the *n*-direction ($n = 1, 2, 3$).

CASUC Field: no History: yes .fil: no .dat: yes

Connector accumulated slip in the instantaneous slip direction. Available only if friction is defined in the slip direction.

CIVC Field: no History: yes .fil: yes .dat: yes

Connector instantaneous velocity in the slip direction. Available only if friction is defined in the slip direction.

CRF Field: no History: yes .fil: yes .dat: yes

All components of connector reaction forces and moments.

CRF*n* Field: no History: yes .fil: no .dat: yes

Connector reaction force component *n* ($n = 1, 2, 3$).

CRM*n* Field: no History: yes .fil: no .dat: yes

Connector reaction moment component *n* ($n = 1, 2, 3$).

CCF Field: no History: yes .fil: yes .dat: yes

All components of connector concentrated forces and moments.

CCF*n* Field: no History: yes .fil: no .dat: yes

Connector concentrated force component *n* ($n = 1, 2, 3$).

CCM*n* Field: no History: yes .fil: no .dat: yes

Connector concentrated moment component *n* ($n = 1, 2, 3$).

CP Field: no History: yes .fil: yes .dat: yes

Relative positions in all directions.

CP*n* Field: no History: yes .fil: no .dat: yes

Relative position in the *n*-direction ($n = 1, 2, 3$).

CPR*n* Field: no History: yes .fil: no .dat: yes

Relative angular position in the *n*-direction ($n = 1, 2, 3$).

CU Field: yes History: yes .fil: yes .dat: yes

Relative displacements and rotations in all directions.

CU*n* Field: no History: yes .fil: no .dat: yes

Relative displacement in the *n*-direction ($n = 1, 2, 3$).

CUR*n* Field: no History: yes .fil: no .dat: yes

Relative rotation in the *n*-direction ($n = 1, 2, 3$).

CCU Field: no History: yes .fil: yes .dat: yes

Constitutive displacements and rotations in all directions.

CCU*n* Field: no History: yes .fil: no .dat: yes

Constitutive displacement in the *n*-direction ($n = 1, 2, 3$).

CCUR*n* Field: no History: yes .fil: no .dat: yes

Constitutive rotation in the *n*-direction ($n = 1, 2, 3$).

CV Field: no History: yes .fil: yes .dat: yes

Relative velocities in all directions.

CV*n* Field: no History: yes .fil: no .dat: yes

Relative velocity in the *n*-direction ($n = 1, 2, 3$).

CVR*n* Field: no History: yes .fil: no .dat: yes

Relative angular velocity in the *n*-direction ($n = 1, 2, 3$).

CA Field: no History: yes .fil: yes .dat: yes

Relative accelerations in all directions.

CA*n* Field: no History: yes .fil: no .dat: yes

Relative acceleration in the *n*-direction ($n = 1, 2, 3$).

CAR*n* Field: no History: yes .fil: no .dat: yes

Relative angular acceleration in the *n*-direction ($n = 1, 2, 3$).

CFAILST Field: no History: yes .fil: yes .dat: yes

All flags for connector failure status.

CFAILST*i* Field: no History: yes .fil: no .dat: yes

Flag for connector failure status in the *i*-direction ($i = 1, \ldots, 6$).

Shear Panel Elements

SQEQ Field: yes History: yes .fil: no .dat: no

Equivalent shear flow.

Element Face Variables

The output variables listed below are available in Abaqus/Standard.

P Field: yes History: no .fil: no .dat: no

Uniformly distributed pressure load on element faces, including those imported using the PRESS co-simulation field ID. When the pressure is defined using **DLOAD*, the variable name is changed automatically to PDLOAD.

HP Field: yes History: no .fil: no .dat: no

Hydrostatic pressure load on element faces. When the pressure is defined using **DLOAD*, the variable name is changed automatically to HPDLOAD.

TRNOR Field: yes History: no .fil: no .dat: no

Normal component (component along face normal) of traction load on element faces.

TRSHR Field: yes History: no .fil: no .dat: no

Shear component (component along face tangent) of traction load on element faces.

TRVEC Field: yes History: no .fil: no .dat: no

Traction load vector on element faces.

FLUXS Field: yes History: no .fil: no .dat: no

Uniformly distributed heat fluxes on element faces.

FILMCOEF Field: yes History: no .fil: no .dat: no

Reference film coefficient value on element faces.

SINKTEMP Field: yes History: no .fil: no .dat: no

Reference sink temperature on element faces.

AMBIENTTEMP Field: yes History: no .fil: no .dat: no

Reference ambient temperature on element faces from prescribed boundary radiation.

References:

- *Writing Element Output to the Output Database*

Whole Element Energy Density Variables

The output variables listed below are available in Abaqus/Standard.

ELEDEN Field: yes History: no .fil: no .dat: no

All energy density components. A limited number of them are available for direct-solution steady-state dynamic and subspace-based steady-state dynamic analyses. Mode-based steady-state dynamic analyses support computation of kinetic and strain energy densities as well as the density of the energy loss due to viscous and structural damping. In frequency extraction analyses, the values of energy densities in the element are normalized. Normalization is performed for each eigenmode separately, such that the kinetic and strain energies for the whole model add up to one.

EKEDEN Field: yes History: yes .fil: no .dat: no

Kinetic energy density in the element. In steady-state dynamic and frequency extraction analyses, this is the cyclic mean value. In mode-based and direct-solution steady-state dynamic analyses and in frequency extraction analyses, only field output in the output database is supported for this variable. In frequency extraction analyses, the value of kinetic energy density in the element is normalized. Normalization is performed for each eigenmode separately, such that the kinetic and strain energies for the whole model add up to one.

EKEDENA Field: yes History: no .fil: no .dat: no

Kinetic energy density amplitude in the element. This variable is available only in mode-based and direct-solution steady-state dynamics analyses.

EKEDENP Field: yes History: no .fil: no .dat: no

Kinetic energy density peak value in the element. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ESEDEN Field: yes History: yes .fil: no .dat: no

Total elastic strain energy density in the element. When the Mullins effect is modeled with hyperelastic materials, this quantity represents only the recoverable part of the energy density in the element. In steady-state dynamic and frequency extraction analyses, this is the cyclic mean value. In mode-based and direct-solution steady-state dynamic analyses and in frequency extraction analyses, only field output in the output database is supported for this variable. In frequency extraction analyses, the value of total elastic strain energy density in the element is normalized. Normalization is performed for each eigenmode separately, such that the kinetic and strain energies for the whole model add up to one.

ESEDENA Field: yes History: no .fil: no .dat: no

Total elastic strain energy density amplitude in the element. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ESEDENP Field: yes History: no .fil: no .dat: no

Total elastic strain energy density peak value in the element. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

EPDDEN Field: yes History: yes .fil: no .dat: no

Total energy dissipated per unit volume in the element by rate-independent and rate-dependent plastic deformation. Not available for steady-state dynamic analysis.

ECDDEN Field: yes History: yes .fil: no .dat: no

Total energy dissipated per unit volume in the element by creep, swelling, and viscoelasticity. Not available for steady-state dynamic analysis.

EVDDEN Field: yes History: yes .fil: no .dat: no

Total energy dissipated per unit volume in the element by viscous effects, not inclusive of energy dissipated through static stabilization or viscoelasticity. In mode-based and direct-solution steady-state dynamic analyses, only field output in the output database is supported for this variable.

EVDDENE Field: yes History: no .fil: no .dat: no

Total energy dissipated per unit volume in the element by viscous effects due to the material damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

EVDDENG Field: yes History: no .fil: no .dat: no

Total energy dissipated per unit volume in the element by viscous effects due to the global damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

EHDDEN Field: yes History: no .fil: no .dat: no

Total energy dissipated per unit volume in the element due to structural damping. This variable includes energy loss due to the material and global structural damping and is available only in mode-based and direct-solution steady-state dynamic analyses.

EHDDENE Field: yes History: no .fil: no .dat: no

Total energy dissipated per unit volume in the element due to the material structural damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

EHDDENG Field: yes History: no .fil: no .dat: no

Total energy dissipated per unit volume in the element due to the global structural damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ESDDEN Field: yes History: yes .fil: no .dat: no

Total energy dissipated per unit volume in the element resulting from static stabilization. Not available for steady-state dynamic analysis.

ECTEDEN Field: yes History: yes .fil: no .dat: no

Total electrostatic energy density in the element. Not available for steady-state dynamic analysis.

EASEDEN Field: yes History: yes .fil: no .dat: no

Total “artificial” strain energy density in the element (energy associated with constraints used to remove singular modes, such as hourglass control, and with constraints used to make the drill rotation follow the in-plane rotation of the shell element). Not available for steady-state dynamic analysis.

EDMDDEN Field: yes History: yes .fil: no .dat: no

Total energy dissipated per unit volume in the element by damage. Not available for steady-state dynamic analysis.

References:

- *Energy balance*

Whole Element Error Indicator Variables

The output variables listed below are available in Abaqus/Standard.

ENDEN Field: yes History: no .fil: no .dat: no

Element energy density, including plastic dissipation and creep dissipation if present.

ENDENERI Field: yes History: no .fil: no .dat: no

Element energy density error indicator, including plastic dissipation error and creep dissipation error if present.

MISESAVG Field: yes History: no .fil: no .dat: no

Element average Mises equivalent stress.

MISESERI Field: yes History: no .fil: no .dat: no

Element Mises equivalent stress error indicator.

PEEQAVG Field: yes History: no .fil: no .dat: no

Element average equivalent plastic strain.

PEEQERI Field: yes History: no .fil: no .dat: no

Element equivalent plastic strain error indicator.

PEAVG Field: yes History: no .fil: no .dat: no

Element average plastic strain.

PEERI Field: yes History: no .fil: no .dat: no

Element plastic strain error indicator.

CEAVG Field: yes History: no .fil: no .dat: no

Element average creep strain.

CEERI Field: yes History: no .fil: no .dat: no

Element creep strain error indicator.

HFLAVG Field: yes History: no .fil: no .dat: no

Element average heat flux.

HFLERI Field: yes History: no .fil: no .dat: no

Element heat flux error indicator.

EFLAVG Field: yes History: no .fil: no .dat: no

Element average electric flux.

EFLERI Field: yes History: no .fil: no .dat: no

Element electric flux error indicator.

EPGAVG Field: yes History: no .fil: no .dat: no

Element average electric potential gradient.

EPGERI

Field: yes History: no .fil: no .dat: no

Element electric potential gradient error indicator.

References:

- *Selection of Error Indicators Influencing Adaptive Remeshing*

Nodal Variables

The output variables listed below are available in Abaqus/Standard.

U Field: yes History: yes .fil: yes .dat: yes

All physical displacement components, including rotations at nodes with rotational degrees of freedom (for output to the output database, only field-type output includes the rotations).

UACT Field: yes History: yes .fil: no .dat: no

All physical displacement components, including rotations at nodes with rotational degrees of freedom (for output to the output database, only field-type output includes the rotations), measured from the time the node is activated.

URACT Field: yes History: yes .fil: no .dat: no

All rotational displacement components measured from the time the node is activated.

UTACT Field: yes History: yes .fil: no .dat: no

All translational displacement components measured from the time the node is activated.

UT Field: yes History: yes .fil: no .dat: no

All translational displacement components.

UR Field: yes History: yes .fil: no .dat: yes

All rotational displacement components.

U*n* Field: no History: yes .fil: no .dat: yes

u_n displacement component ($n = 1, 2, 3$).

UR*n* Field: no History: yes .fil: no .dat: yes

ϕ_n rotation component ($n = 1, 2, 3$).

WARP Field: yes History: yes .fil: no .dat: yes

Warping magnitude. Available only for open-section beam elements.

V Field: yes History: yes .fil: yes .dat: yes

All velocity components, including rotational velocities at nodes with rotational degrees of freedom (for output to the output database, only field-type output includes the rotational velocities).

VT Field: yes History: yes .fil: no .dat: no

All translational velocity components.

VR Field: yes History: yes .fil: no .dat: yes

All rotational velocity components.

V*n* Field: no History: yes .fil: no .dat: yes

$\dot{u}_n$ velocity component ($n = 1, 2, 3$).

VR*n* Field: no History: yes .fil: no .dat: yes

$\dot{\phi}_n$ rotational velocity component ($n = 1, 2, 3$).

A Field: yes History: yes .fil: yes .dat: yes

All acceleration components, including rotational accelerations at nodes with rotational degrees of freedom (for output to the output database, only field-type output includes the rotational accelerations).

AT Field: yes History: yes .fil: no .dat: no

All translational acceleration components.

AR Field: yes History: yes .fil: no .dat: yes

All rotational acceleration components.

A*n* Field: no History: yes .fil: no .dat: yes

$\ddot{u}_n$ acceleration component ($n = 1, 2, 3$).

AR*n* Field: no History: yes .fil: no .dat: yes

$\ddot{\phi}_n$ rotational acceleration component ($n = 1, 2, 3$).

POR Field: yes History: yes .fil: yes .dat: yes

Pore or acoustic pressure at a node.

SLURRYVF Field: yes History: yes .fil: yes .dat: yes

Slurry concentration (volumetric fraction of particles in a slurry flow) at a node.

CFF Field: yes History: yes .fil: yes .dat: yes

Concentrated fluid flow at a node, including those imported using the CFLOW co-simulation field ID.

NT Field: yes History: yes .fil: yes .dat: yes

All temperature values at a node, including those imported using the TEMP co-simulation field ID. These will be the temperatures defined as degrees of freedom if heat transfer elements are connected to the node, or predefined temperatures if the node is connected only to stress or mass diffusion elements without temperature degrees of freedom.

NT*n* Field: no History: yes .fil: no .dat: yes

Temperature degree of freedom *n* at a node ($n = 11, 12, \ldots$).

EPOT Field: yes History: yes .fil: yes .dat: yes

All electrical potential degrees of freedom at a node.

EPOTE Field: yes History: yes .fil: yes .dat: yes

All fluid electrical potential degrees of freedom at a node.

NNC Field: yes History: yes .fil: yes .dat: yes

All normalized concentration values at a node.

NNC*n* Field: no History: yes .fil: no .dat: yes

Normalized concentration degree of freedom *n* at a node ($n = 11$).

NNCE Field: yes History: yes .fil: yes .dat: yes

Ion concentration degree of freedom at a node.

NNCS Field: yes History: yes .fil: yes .dat: yes

Species concentration degree of freedom at a node.

RF Field: yes History: yes .fil: yes .dat: yes

All components of reaction forces, including components of reaction moments at nodes with rotational degrees of freedom (conjugate to prescribed displacements and rotations). For output to the output database, only the field-type output includes the components of reaction moments at nodes with rotational degrees of freedom.

RT Field: yes History: yes .fil: no .dat: no

All reaction force components.

RM Field: yes History: yes .fil: no .dat: yes

All reaction moment components.

RF*n* Field: no History: yes .fil: no .dat: yes

Reaction force component *n* ($n = 1, 2, 3$) (conjugate to prescribed displacement u_n).

RM*n* Field: no History: yes .fil: no .dat: yes

Reaction moment component *n* ($n = 1, 2, 3$) (conjugate to prescribed rotation ϕ_n).

RWM Field: yes History: yes .fil: no .dat: yes

Reaction bimoment in degree of freedom 7, conjugate to prescribed warping amplitude. Available only for open-section beam elements.

CF Field: yes History: yes .fil: yes .dat: yes

All components of point loads and concentrated moments, including loads imported using the CF co-simulation field ID.

CF*n* Field: no History: yes .fil: no .dat: yes

Point load component *n* ($n = 1, 2, 3$).

CM*n* Field: no History: yes .fil: no .dat: yes

Point moment component *n* ($n = 1, 2, 3$).

CW Field: no History: yes .fil: no .dat: yes

Load component in degree of freedom 7. Available only for open-section beam elements.

TF Field: yes History: yes .fil: yes .dat: yes

All components of total forces, including components of total moments at nodes with rotational degrees of freedom. Total force is the sum of the reaction force and point loads. For output to the output database, only the field-type output includes the components of total moments at nodes with rotational degrees of freedom.

TF*n* Field: no History: yes .fil: no .dat: yes

Total force component *n* ($n = 1, 2, 3$).

TM*n* Field: no History: yes .fil: no .dat: yes

Total moment component *n* ($n = 1, 2, 3$).

VF Field: yes History: yes .fil: yes .dat: yes

All components of viscous forces and moments due to static stabilization.

VFn Field: no History: yes .fil: no .dat: yes

Stabilization viscous force component *n* ($n = 1, 2, 3$).

VMn Field: no History: yes .fil: no .dat: yes

Stabilization viscous moment component *n* ($n = 1, 2, 3$).

COORD Field: yes History: yes .fil: yes .dat: yes

Coordinates of the node. These are the current coordinates if the large-displacement formulation is being used.

COORn Field: no History: yes .fil: no .dat: yes

Coordinate *n* ($n = 1, 2, 3$).

STRAINFREE Field: yes History: no .fil: no .dat: no

Strain-free adjustments to initial nodal positions (adjusted position minus unadjusted position; only written to the output database (`.odb`) file for the original field output frame at zero time).

RCHG Field: yes History: yes .fil: yes .dat: yes

Reactive electrical nodal charge (conjugate to prescribed electrical potential).

CECHG Field: yes History: yes .fil: yes .dat: yes

Concentrated electrical nodal charge.

RECUR Field: yes History: yes .fil: yes .dat: yes

Reactive electrical nodal current (conjugate to prescribed electrical potential).

RECURE Field: yes History: yes .fil: yes .dat: yes

Reactive fluid electrical nodal current (conjugate to prescribed electrical potential for the fluid).

CECUR Field: yes History: yes .fil: yes .dat: yes

Concentrated electrical nodal current.

PCAV Field: no History: yes .fil: yes .dat: yes

Hydrostatic fluid gauge pressure (total pressure = ambient pressure + hydrostatic fluid gauge pressure).

CVOL Field: no History: yes .fil: yes .dat: yes

Hydrostatic fluid cavity volume.

MOT Field: yes History: yes .fil: yes .dat: yes

All components of motion in cavity radiation heat transfer analysis.

MOTn Field: no History: yes .fil: no .dat: yes

m_n motion component ($n = 1, 2, 3$) in cavity radiation heat transfer analysis.

STIFN Field: yes History: no .fil: no .dat: no

Local normal stiffness. Defined as the ratio of the normal force and normal displacement magnitudes at a surface node, where the force is statically applied at the surface node in the normal direction to the surface. Available only in frequency extraction procedures.

References:

- *Node Output*
- *Writing Nodal Output to the Output Database*

Acoustic Quantities

POR Field: yes History: yes .fil: yes .dat: yes

Acoustic pressure.

INFR Field: yes History: no .fil: no .dat: no

Acoustic infinite element “radius,” used in the coordinate map for these elements. Available only if the direct-solution steady-state dynamic procedure is used, and available only for nodes attached to acoustic infinite elements.

INFC Field: yes History: no .fil: no .dat: no

Acoustic infinite element “cosine,” used in the coordinate map for these elements. Available only if the direct-solution steady-state dynamic procedure is used, and available only for nodes attached to acoustic infinite elements.

INFN Field: yes History: no .fil: no .dat: no

Acoustic infinite element normal vector. Available only if the direct-solution steady-state dynamic procedure is used, and available only for nodes attached to acoustic infinite elements.

PINF Field: yes History: no .fil: no .dat: no

Acoustic pressure coefficients for the higher-order basis functions in acoustic infinite elements. Available only if the direct-solution steady-state dynamic procedure is used, and available only for acoustic infinite elements.

SPL Field: yes History: yes .fil: no .dat: no

Acoustic sound pressure level at a node.

Enriched Element Quantities

PHILSM Field: yes History: yes .fil: no .dat: no

Signed distance function to describe the crack surface.

PSILSM Field: yes History: yes .fil: no .dat: no

Signed distance function to describe the initial crack front.

Fracture Mechanics Quantities

J Field: yes History: no .fil: no .dat: no

Averaged value of the contour integral evaluated based on the conventional finite element method.

K Field: yes History: no .fil: no .dat: no

Averaged values of the stress intensity factors evaluated based on the conventional finite element method.

TSTRESS Field: yes History: no .fil: no .dat: no

Averaged value of the T-stress evaluated based on the conventional finite element method.

Heat or Mass Flux or Ion Flux

The following variables correspond to heat flux in temperature analyses or concentration volumetric flux in mass diffusion analysis:

RFL Field: yes History: yes .fil: yes .dat: yes

All reaction flux values (conjugate to prescribed temperature or normalized concentration).

RFL*n* Field: no History: yes .fil: no .dat: yes

Reaction flux value *n* at a node ($n = 11, 12, \ldots$) (conjugate to prescribed temperature or normalized concentration).

RFLCE Field: yes History: yes .fil: yes .dat: yes

All reaction flux values (conjugate to prescribed ion concentration).

RFLCS Field: yes History: yes .fil: yes .dat: yes

All reaction flux values (conjugate to prescribed species concentration).

CFL Field: yes History: yes .fil: yes .dat: yes

All concentrated flux values, including those imported using the CFL co-simulation field ID.

CFL*n* Field: no History: yes .fil: no .dat: yes

Concentrated flux values *n* at a node ($n = 11, 12, \ldots$).

RFLE Field: yes History: yes .fil: yes .dat: yes

The total flux at the node (including flux convected through the node in convection elements), excluding external fluxes (due to concentrated fluxes, distributed fluxes, film conditions, radiation conditions, and radiation view factors). The value of RFLE is, thus, equal and opposite to the sum of all applied fluxes.

RFLE*n* Field: no History: yes .fil: no .dat: yes

Flux value *n* excluding externally applied flux loads at a node ($n = 11, 12, \ldots$).

Steady-State Dynamic Analysis

The following variables are available only for steady-state (frequency domain) dynamic analyses (modal and direct). These variables include both magnitude and phase angle for all components. Phase angles are given in degrees. In the data file there are two lines of output for each request. The first line contains the magnitude, and the second line (indicated by the SSD footnote) contains the phase angle. In the results file, the magnitudes of all components are first, followed by the phase angles of all components.

PU Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of all displacement components at the node and magnitude and phase angle of the rotations at nodes with rotational degrees of freedom.

PU*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of component *n* of the displacement ($n = 1, 2, 3$).

PUR*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of component *n* of the rotation ($n = 1, 2, 3$).

PPOR Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of the fluid, pore, or acoustic pressure at the node.

PHPOT Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of the electrical potential at the node.

PRF Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of the reaction forces at the node and of the reaction moments at nodes with rotational degrees of freedom.

PRF*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of component *n* of the reaction force ($n = 1, 2, 3$).

PRM*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of component *n* of the reaction moment ($n = 1, 2, 3$).

PHCHG Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of the reactive charge at the node.

VN Field: yes History: yes .fil: no .dat: no

Complex-valued surface normal velocity. Defined as the projection of the nodal complex velocity vector on the local (real-valued) surface normal vector at the node.

VNSQ Field: yes History: yes .fil: no .dat: no

Real-valued surface normal velocity squared. Defined as the scalar (real) magnitude squared of the surface normal velocity vector.

AVNSQ Field: yes History: yes .fil: no .dat: no

Area-weighted surface normal velocity squared or acoustic power normalized by the acoustic impedance of the surrounding fluid. A real-valued scalar equal to VNSQ multiplied by the local surface area adjacent to the node.

ERPAC Field: yes History: yes .fil: no .dat: no

Complex-valued acoustic pressure on the radiating element-based structural surface. This variable is derived from the equivalent radiated power ERPWR.

ERPWR Field: yes History: yes .fil: no .dat: no

Equivalent radiated power on an element-based structural surface. A real-valued scalar equal to AVNSQ multiplied by the acoustic impedance.

ERPWRDEN Field: yes History: yes .fil: no .dat: no

Equivalent radiated power density on an element-based structural surface. A real-valued scalar equal to VNSQ multiplied by the acoustic impedance.

Modal Dynamic, Steady-State, and Random Response Analysis

The following variables are available only for modal dynamic, steady-state (frequency domain), and random response analyses. "Relative" values are measured relative to the motion of the primary base and are obtained with the identifiers *U*, *V*, and *A*; "Total" values include the motion of the primary base. For steady-state dynamic output printed to the data file, there are two lines printed for each request; the first line contains the real part of the variable, and the second line (indicated by the SSD footnote) contains the imaginary part.

TU Field: yes History: yes .fil: yes .dat: yes

All components of the total displacements at the node and of the total rotations at nodes with rotational degrees of freedom.

TU*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the total displacement ($n = 1, 2, 3$).

TUR*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the total rotation ($n = 1, 2, 3$).

TV Field: yes History: yes .fil: yes .dat: yes

All components of the total velocity at the node, including rotational velocities at nodes with rotational degrees of freedom.

TV*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the total velocity ($n = 1, 2, 3$).

TVR*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the total rate of rotation ($n = 1, 2, 3$).

TA Field: yes History: yes .fil: yes .dat: yes

All components of the total acceleration at the node, including rotational accelerations at nodes with rotational degrees of freedom.

TA*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the total acceleration ($n = 1, 2, 3$).

TAR*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the total rotational acceleration ($n = 1, 2, 3$).

Mode-Based Steady-State Dynamic Analysis

The following variables are available only for steady-state (frequency domain) dynamic analysis based on modal superposition. “Total” values include the base motion.

PTU Field: no History: no .fil: yes .dat: yes

Magnitude and phase angle of the total displacement components at the node and magnitude and phase angle of the total rotations at nodes with rotational degrees of freedom.

PTU*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of component *n* of the total displacement ($n = 1, 2, 3$).

PTUR*n* Field: no History: no .fil: no .dat: yes

Magnitude and phase angle of component *n* of the total rotation ($n = 1, 2, 3$).

Pore Pressure Analysis

The following variables correspond to fluid volume flux in pore pressure analyses.

RVF Field: yes History: yes .fil: yes .dat: yes

Reaction fluid volume flux due to prescribed pressure. This flux is the rate at which fluid volume is entering or leaving the model through the node to maintain the prescribed pressure boundary condition. A positive value of RVF indicates fluid is entering the model.

RVT Field: yes History: yes .fil: yes .dat: yes

Reaction total fluid volume (computed only in a transient coupled pore fluid diffusion/stress analysis). This value is the time integrated value of RVF.

FLDVEL Field: yes History: yes .fil: yes .dat: yes

Nodal fluid velocity.

SVVF Field: yes History: yes .fil: no .dat: no

Stabilized viscous fluid volume flux at nodes due to the projection of pressure into the strain space.

Random Response Analysis

The following variables are available only for random response dynamic analysis. “Relative” values are measured relative to the base motion; “Total” values include the base motion.

RU Field: yes History: yes .fil: yes .dat: yes

Root mean square values of all components of the relative displacement at the node and of the components of rotation at nodes with rotational degrees of freedom.

RU*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the relative displacement ($n = 1, 2, 3$).

RUR*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the relative rotation ($n = 1, 2, 3$).

RTU Field: yes History: yes .fil: yes .dat: yes

Root mean square values of all components of the total displacement at the node and of the components of total rotation at nodes with rotational degrees of freedom.

RTU*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the total displacement ($n = 1, 2, 3$).

RTUR*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the total rotation ($n = 1, 2, 3$).

RV Field: yes History: yes .fil: yes .dat: yes

Root mean square values of all components of the relative velocity at the node and of the components of the rate of rotation at nodes with rotational degrees of freedom.

RV*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the relative velocity ($n = 1, 2, 3$).

RVR*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the relative rate of rotation ($n = 1, 2, 3$).

RTV Field: yes History: yes .fil: yes .dat: yes

Root mean square values of all components of the total velocity at the node and of the components of total rotation at nodes with rotational degrees of freedom.

RTV*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the total velocity ($n = 1, 2, 3$).

RTVR*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the total rate of rotation ($n = 1, 2, 3$).

RA Field: yes History: yes .fil: yes .dat: yes

Root mean square values of all components of the relative acceleration at the node and of the components of rotational acceleration at nodes with rotational degrees of freedom.

RA*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the relative acceleration ($n = 1, 2, 3$).

RAR*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the relative rotational acceleration ($n = 1, 2, 3$).

RTA Field: yes History: yes .fil: yes .dat: yes

Root mean square values of all components of the total acceleration at the node and of the components of rotational acceleration at nodes with rotational degrees of freedom.

RTA*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component *n* of the total value of acceleration ($n = 1, 2, 3$).

RTAR*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component n of the total rotational acceleration ($n = 1, 2, 3$).

RRF Field: yes History: yes .fil: yes .dat: yes

Root mean square values of all components of the reaction forces and of reaction moments at nodes with rotational degrees of freedom.

RRF*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component n of the reaction force ($n = 1, 2, 3$).

RRM*n* Field: no History: yes .fil: no .dat: yes

Root mean square value of component n of the reaction moment ($n = 1, 2, 3$).

Steady-State Transport Analysis

The following variables are available only for steady-state transport analysis:

SSTIF Field: yes History: yes .fil: no .dat: no

Nodal forces due to steady-state transport inertia loading. This output allows you to visualize the inertial forces generated in a deformable spinning wheel.

SSTSF Field: yes History: yes .fil: no .dat: no

Nodal forces due to inertia stabilization in a steady-state transport analysis. This output allows you to visualize the artificial inertia stabilization loads and compare them to the inertial forces that output variable SSTIF generated in a deformable spinning wheel.

SSTIRF Field: no History: yes .fil: no .dat: no

Resultant of all inertia nodal loads in a steady-state transport analysis. The node set prescribed for this history output request must have only one member. Because the output is a resultant of forces, the node chosen for output does not affect the computation. Typically, output is requested on a reference node on the axle of a wheel.

SSTSRF Field: no History: yes .fil: no .dat: no

Resultant of all inertia stabilization loads in a steady-state transport analysis. The node set prescribed for this history output request must have only one member. Because the output is a resultant of forces, the node chosen for output does not affect the computation. Typically, output is requested on a reference node on the axle of a wheel. This output allows you to compare the relative magnitudes of the resultant inertia nodal loads (output variable SSTIRF) to that of the artificial inertia stabilization loads.

SSTIRM Field: no History: yes .fil: no .dat: no

Resultant moment of all inertia nodal loads in a steady-state transport analysis. The node set prescribed for this history output request must have only one member. Because the output is a resultant moment, the node chosen for output does affect the computation. Typically, output is requested on a reference node on the axle of a wheel.

SSTSRM Field: no History: yes .fil: no .dat: no

Resultant moment of all inertia stabilization loads in a steady-state transport analysis. The node set prescribed for this history output request must have only one member. Because the output is a resultant moment, the node chosen for output does affect the computation. Typically, output is requested on a reference node on the axle of a wheel. This output allows you to compare the relative magnitudes of the resultant moment due to inertia nodal loads (output variable SSTIRM) to that of the artificial inertia stabilization loads. Understanding this value is particularly important while looking for a free-rolling solution of a tire in which a zero reaction moment about the axle is sought.

One-Step Inverse Analysis

The following variable is available only for one-step inverse analysis:

UINV Field: yes History: yes .fil: no .dat: no

Inverse displacement. The displacement from the specified final deformed configuration to the initial configuration.

Modal Variables

The output variables listed below are available in Abaqus/Standard.

GU Field: no History: yes .fil: yes .dat: yes

Generalized displacements for all modes.

GU*n* Field: no History: yes .fil: no .dat: yes

Generalized displacement for mode *n*.

GV Field: no History: yes .fil: yes .dat: yes

Generalized velocities for all modes.

GV*n* Field: no History: yes .fil: no .dat: yes

Generalized velocity for mode *n*.

GA Field: no History: yes .fil: yes .dat: yes

Generalized acceleration for all modes.

GA*n* Field: no History: yes .fil: no .dat: yes

Generalized acceleration for mode *n*.

GPU Field: no History: yes .fil: yes .dat: yes

Phase angle of generalized displacements for all modes.

GPU*n* Field: no History: yes .fil: no .dat: yes

Phase angle of generalized displacement for mode *n*.

GPV Field: no History: yes .fil: yes .dat: yes

Phase angle of generalized velocities for all modes.

GPV*n* Field: no History: yes .fil: no .dat: yes

Phase angle of generalized velocity for mode *n*.

GPA Field: no History: yes .fil: yes .dat: yes

Phase angle of generalized acceleration for all modes.

GPA*n* Field: no History: yes .fil: no .dat: yes

Phase angle of generalized acceleration for mode *n*.

SNE Field: no History: yes .fil: yes .dat: yes

Elastic strain energy for the entire model per each mode (not available for random response analysis).

SNE*n* Field: no History: yes .fil: no .dat: yes

Elastic strain energy for the entire model for mode *n* (not available for random response analysis).

KE Field: no History: yes .fil: yes .dat: yes

Kinetic energy for the entire model per each mode (not available for random response analysis).

KE*n* Field: no History: yes .fil: no .dat: yes

Kinetic energy for the entire model for mode *n* (not available for random response analysis).

T Field: no History: yes .fil: yes .dat: yes

External work for the entire model per each mode (not available for random response analysis).

T*n* Field: no History: yes .fil: no .dat: yes

External work for the entire model for mode *n* (not available for random response analysis).

BM Field: no History: yes .fil: yes .dat: yes

Base motion (not available for random response or response spectrum analyses).

References:

- *Modal Output from Abaqus/Standard*
- *Modal Output from Abaqus/Standard*

Surface Variables

The output variables listed below are available in Abaqus/Standard.

References:

- *Surface Output from Abaqus/Standard*
- *Writing Surface Output to the Output Database*
- *About Contact Pairs in Abaqus/Standard*
- *Contact Property Models*

Mechanical Analysis–Nodal Quantities

CSTRESS Field: yes History: yes .fil: yes .dat: yes

Contact pressure (CPRESS) and frictional shear stresses (CSHEAR). Output is also available on the main surface to the `.odb` file in a single main-secondary setting.

CSTRESSETOS Field: yes History: no .fil: no .dat: no

Contact pressure (CPRESSETOS) and frictional shear stresses (CSHEARETOS) due to edge-to-surface contact constraints. Output is also available on the main surface to the `.odb` file in a single main-secondary setting.

CSTRESSERI Field: yes History: no .fil: no .dat: no

Error indicators for the contact pressure (CPRESSERI) and frictional shear stresses (CSHEARERI). Output is also available on the main surface to the `.odb` file in a single main-secondary setting.

CDSTRESS Field: yes History: yes .fil: yes .dat: yes

Viscous pressure (CDPRESS) and viscous shear stresses (CDSHEAR). Output is also available on the main surface to the `.odb` file in a single main-secondary setting.

CDISP Field: yes History: yes .fil: yes .dat: yes

Contact opening (COPEN) and relative tangential motions (CSLIP).

CSLIPEQ Field: yes History: no .fil: no .dat: no

Contact equivalent relative tangential motion.

CSL_NORMALIZED Field: yes History: no .fil: no .dat: no

Normalized contact equivalent relative tangential motion for the small-sliding tracking approach in general contact.

CSLIP_PL Field: yes History: no .fil: no .dat: no

Contact plastic relative tangential motions (CSLIP_PL1 and CSLIP_PL2).

CSLIP_PLEQ Field: yes History: no .fil: no .dat: no

Contact equivalent plastic relative tangential motion.

CDISPETOS Field: yes History: no .fil: no .dat: no

Contact opening (COPENETOS) and relative tangential motions (CSLIPETOS) for edge-to-surface contact constraints.

CFORCE Field: yes History: no .fil: no .dat: no

Contact normal force (CNORMF) and frictional shear force (CSHEARF). Output is also available on the main surface to the `.odb` file in a single main-secondary setting.

CLINELOAD Field: yes History: no .fil: no .dat: no

Contact load due to line contact from edge-to-surface and radial edge-to-edge constraints in units of force per length. The normal (CLINELOADN) and frictional shear (CLINELOADT) components are available only for general contact to the `.odb` file.

CNAREA Field: yes History: no .fil: no .dat: no

Contact nodal area for each node in contact or near an opposing surface, calculated as $A_i = \Sigma_{j=1}^{n_{cont\_i}} |c_{ij}| A_j$, where n_{cont_i} is the number of potential contact constraints involving node i, A_j is the area of constraint j, and c_{ij} is a contact constraint coefficient.

CPOINTLOAD Field: yes History: no .fil: no .dat: no

Contact load in units of force due to point contact from edge-to-edge constraints using the cross formulation. The normal (CPOINTLOADN) and frictional shear (CPOINTLOADT) components are available only for general contact to the `.odb` file.

CRKDISP Field: yes History: yes .fil: no .dat: no

Crack opening and relative tangential motions on cracked surfaces in enriched elements.

CRKSTRESS Field: yes History: yes .fil: no .dat: no

Remaining residual pressure and tangential shear stresses on cracked surfaces in enriched elements.

CSTATUS Field: yes History: no .fil: no .dat: no

Contact status. Output is also available on the main surface to the `.odb` file in a single main-secondary setting.

CSMAXSCRT Field: yes History: yes .fil: no .dat: no

Maximum value of the maximum stress-based damage initiation criterion for cohesive contact up to the current increment.

CSQUADSCRT Field: yes History: yes .fil: no .dat: no

Maximum value of the quadratic stress-based damage initiation criterion for cohesive contact up to the current increment.

CSMAXUCRT Field: yes History: yes .fil: no .dat: no

Maximum value of the maximum separation-based damage initiation criterion for cohesive contact up to the current increment.

CSQUADUCRT Field: yes History: yes .fil: no .dat: no

Maximum value of the quadratic separation-based damage initiation criterion for cohesive contact up to the current increment.

CSDMG Field: yes History: yes .fil: no .dat: no

Damage variable for cohesive surfaces or for cracked surfaces in enriched elements.

CTANDIR Field: yes History: no .fil: no .dat: no

Instantaneous contact tangent directions (CTANDIR1 and CTANDIR2).

CWEAR Field: yes History: yes .fil: no .dat: no

Accumulated scalar nodal contact wear distance.

PPRESS Field: yes History: yes .fil: yes .dat: yes

Pressure due to fluid pressure penetration loading.

PFORCE Field: yes History: no .fil: no .dat: no

Force due to fluid pressure penetration loading.

SDV Field: yes History: yes .fil: yes .dat: yes

Solution-dependent state variables.

Mechanical Analysis–Whole Surface Quantities

CFN Field: no History: yes .fil: yes .dat: yes

Total force due to contact pressure (CFN*n*, *n* = 1, 2, 3).

CFNM Field: no History: yes .fil: no .dat: no

Magnitude of total force due to contact pressure.

CFS Field: no History: yes .fil: yes .dat: yes

Total force due to frictional stress (CFS*n*, *n* = 1, 2, 3).

CFSM Field: no History: yes .fil: no .dat: no

Magnitude of total force due to frictional stress.

CFT Field: no History: yes .fil: yes .dat: yes

Total force due to contact pressure and frictional stress (CFT*n*, *n* = 1, 2, 3).

CFTM Field: no History: yes .fil: no .dat: no

Magnitude of total force due to contact pressure and frictional stress.

CICPS Field: no History: yes .fil: no .dat: no

The scalar integration of the contact pressure over the surface.

CMN Field: no History: yes .fil: yes .dat: yes

Total moment about the origin due to contact pressure (CMN*n*, *n* = 1, 2, 3).

CMNM Field: no History: yes .fil: no .dat: no

Magnitude of total moment about origin due to contact pressure.

CMS Field: no History: yes .fil: yes .dat: yes

Total moment about the origin due to frictional stress (CMS*n*, *n* = 1, 2, 3).

CMSM Field: no History: yes .fil: no .dat: no

Magnitude of total moment about the origin due to frictional stress.

CMT Field: no History: yes .fil: yes .dat: yes

Total moment about the origin due to contact pressure and frictional stress (CMT*n*, *n* = 1, 2, 3).

CMTM Field: no History: yes .fil: no .dat: no

Magnitude of total moment about the origin due to contact pressure and frictional stress.

CAREA Field: no History: yes .fil: yes .dat: yes

Total area in contact.

CTRQ Field: no History: yes .fil: yes .dat: yes

Maximum torque that can be transmitted about the *z*-axis by a contact surface in an axisymmetric analysis with a friction coefficient of unity.

XN Field: no History: yes .fil: yes .dat: yes

Center of the total force due to contact pressure (XN*n*, *n* = 1, 2, 3).

XS Field: no History: yes .fil: yes .dat: yes

Center of the total force due to frictional stress (XS*n*, *n* = 1, 2, 3).

PFN Field: no History: yes .fil: yes .dat: yes

Total force due to fluid pressure penetration loading (PFN*n*, *n* = 1, 2, 3).

XT Field: no History: yes .fil: yes .dat: yes

Center of the total force due to contact pressure and frictional stress (XT*n*, *n* = 1, 2, 3).

Heat Transfer Analysis

HFL Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area leaving the secondary surface.

HFLA Field: yes History: yes .fil: yes .dat: yes

HFL multiplied by the nodal area.

HTL Field: yes History: yes .fil: yes .dat: yes

Time integrated HFL.

HTLA Field: yes History: yes .fil: yes .dat: yes

Time integrated HFLA.

Coupled Thermal-Electrical Analysis

ECD Field: yes History: yes .fil: yes .dat: yes

Electrical current per unit area.

ECDA Field: yes History: yes .fil: yes .dat: yes

ECD multiplied by the nodal area.

ECDT Field: yes History: yes .fil: yes .dat: yes

Time integrated ECD.

ECDTA Field: yes History: yes .fil: yes .dat: yes

Time integrated ECDA.

HFL Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area leaving the secondary surface.

HFLA Field: yes History: yes .fil: yes .dat: yes

HFL multiplied by the nodal area.

HTL Field: yes History: yes .fil: yes .dat: yes

Time integrated HFL.

HTLA Field: yes History: yes .fil: yes .dat: yes

Time integrated HFLA.

SJD Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area due to electrical current.

SJDA Field: yes History: yes .fil: yes .dat: yes

SJD multiplied by the nodal area.

SJDT Field: yes History: yes .fil: yes .dat: yes

Time integrated SJD.

SJDTA Field: yes History: yes .fil: yes .dat: yes

Time integrated SJDA.

WEIGHT Field: yes History: yes .fil: yes .dat: yes

Weighting factor for heat distribution between the interface surfaces.

Fully Coupled Temperature-Displacement Analysis

HFL Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area leaving the secondary surface.

HFLA Field: yes History: yes .fil: yes .dat: yes

HFL multiplied by the nodal area.

HTL Field: yes History: yes .fil: yes .dat: yes

Time integrated HFL.

HTLA Field: yes History: yes .fil: yes .dat: yes

Time integrated HFLA.

SFDR Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area due to frictional dissipation.

SFDRA Field: yes History: yes .fil: yes .dat: yes

SFDR multiplied by the nodal area.

SFDRT Field: yes History: yes .fil: yes .dat: yes

Time integrated SFDR.

SFDRTA Field: yes History: yes .fil: yes .dat: yes

Time integrated SFDRA.

WEIGHT Field: yes History: yes .fil: yes .dat: yes

Weighting factor for heat distribution between the interface surfaces.

Fully Coupled Thermal-Electrical-Structural Analysis

ECD Field: yes History: yes .fil: yes .dat: yes

Electrical current per unit area.

ECDA Field: yes History: yes .fil: yes .dat: yes

ECD multiplied by the nodal area.

ECDT Field: yes History: yes .fil: yes .dat: yes

Time integrated ECD.

ECDTA Field: yes History: yes .fil: yes .dat: yes

Time integrated ECDA.

HFL Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area leaving the secondary surface.

HFLA Field: yes History: yes .fil: yes .dat: yes

HFL multiplied by the nodal area.

HTL Field: yes History: yes .fil: yes .dat: yes

Time integrated HFL.

HTLA Field: yes History: yes .fil: yes .dat: yes

Time integrated HFLA.

SFDR Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area due to frictional dissipation.

SFDRA Field: yes History: yes .fil: yes .dat: yes

SFDR multiplied by the nodal area.

SFDRT Field: yes History: yes .fil: yes .dat: yes

Time integrated SFDR.

SFDRTA Field: yes History: yes .fil: yes .dat: yes

Time integrated SFDRA.

SJD Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area due to electrical current.

SJDA Field: yes History: yes .fil: yes .dat: yes

SJD multiplied by the nodal area.

SJDT Field: yes History: yes .fil: yes .dat: yes

Time integrated SJD.

SJDTA Field: yes History: yes .fil: yes .dat: yes

Time integrated SJDA.

WEIGHT Field: yes History: yes .fil: yes .dat: yes

Weighting factor for heat distribution between the interface surfaces.

Electrical Contact Conductance between Electrolytes

ECDE Field: yes History: yes .fil: yes .dat: yes

Electrical current in electrolyte per unit area.

ECDEA Field: yes History: yes .fil: yes .dat: yes

ECDE multiplied by the nodal area.

ECDET Field: yes History: yes .fil: yes .dat: yes

Time integrated ECDE.

ECDETA Field: yes History: yes .fil: yes .dat: yes

Time integrated ECDEA.

SJDE Field: yes History: yes .fil: yes .dat: yes

Heat flux per unit area due to electrical current in electrolyte.

SJDEA Field: yes History: yes .fil: yes .dat: yes

SJDE multiplied by the nodal area.

SJDET Field: yes History: yes .fil: yes .dat: yes

Time integrated SJDE.

SJDETA Field: yes History: yes .fil: yes .dat: yes

Time integrated SJDEA.

Ion Diffusivity of Electrolytes

MFLE Field: yes History: yes .fil: yes .dat: yes

Ionic flux per unit area leaving the secondary surface.

MFLEA Field: yes History: yes .fil: yes .dat: yes

MFLE multiplied by the nodal area.

MFLET Field: yes History: yes .fil: yes .dat: yes

Time integrated MFLE.

MFLETA Field: yes History: yes .fil: yes .dat: yes

Time integrated MFLEA.

Species Diffusivity in Solids

MFLS
Field: yes History: yes .fil: yes .dat: yes
Species flux per unit area leaving the secondary surface.

MFLSA
Field: yes History: yes .fil: yes .dat: yes
MFLS multiplied by the nodal area.

MFLST
Field: yes History: yes .fil: yes .dat: yes
Time integrated MFLS.

MFLSTA
Field: yes History: yes .fil: yes .dat: yes
Time integrated MFLSA.

Coupled Pore Fluid-Mechanical Analysis–Nodal Quantities

PFL
Field: yes History: yes .fil: yes .dat: yes
Pore fluid volume flux per unit area leaving the secondary surface.

PFLA
Field: yes History: yes .fil: yes .dat: yes
PFL multiplied by the nodal area.

PTL
Field: yes History: yes .fil: yes .dat: yes
Time integrated PFL.

PTLA
Field: yes History: yes .fil: yes .dat: yes
Time integrated PFLA.

Coupled Pore Fluid-Mechanical Analysis–Nodal Quantities in Enriched Elements

GFVR
Field: yes History: yes .fil: no .dat: no
Fluid volume rate within the cracked surfaces in the enriched element.

PORPRES
Field: yes History: yes .fil: no .dat: no
Pore pressure within the cracked surfaces in the enriched element.

PORPRESURF
Field: yes History: yes .fil: no .dat: no
Pore pressure on the cracked surfaces in the enriched element.

LEAKVR
Field: yes History: yes .fil: no .dat: no
Leak-off flow rate on the cracked surfaces in the enriched element.

ALEAKVR
Field: yes History: yes .fil: no .dat: no
Accumulated leak-off flow volume on the cracked surfaces in the enriched element.

Coupled Pore Fluid-Mechanical Analysis–Whole Surface Quantities

TPFL
Field: no History: no .fil: yes .dat: yes

Total pore fluid volume flux leaving the secondary surface.

TPTL
Field: no History: no .fil: yes .dat: yes

Time integrated TPFL.

Coupled Pore Fluid-Mechanical Analysis When Slurry Flow Is Enabled–Nodal Quantities in Enriched Elements

FLVF
Field: yes History: yes .fil: no .dat: no

When multiple fluids are defined, the components of FLVF represent the volume fraction of each fluid type. In particular, FLVF_*i* refers to the volume fraction of fluid type *i*.

SLURRYVF
Field: yes History: yes .fil: no .dat: no

Volumetric concentration of proppant particles in the slurry within the cracked surfaces in the enriched element.

SLURRYAF
Field: yes History: yes .fil: no .dat: no

Concentration of proppant particles in the slurry per unit area within the cracked surfaces in the enriched element.

THKFTCK
Field: yes History: yes .fil: no .dat: no

Filter cake thickness on the top and bottom crack surfaces in the enriched element.

Coupled Pore Fluid-Temperature-Mechanical Analysis–Nodal Quantities in Enriched Elements

PORTEMP
Field: yes History: yes .fil: no .dat: no

Temperature within the cracked surfaces in the enriched element.

Coupled Thermal-Electrochemical-Mechanical Analysis–Surface Facet Quantities

ECHEMQS1
Field: yes History: yes .fil: no .dat: no

Heat generation surface flux of Butler-Volmer electric current, Q_1.

ECHEMQS2
Field: yes History: yes .fil: no .dat: no

Entropic heat generation surface flux, Q_2.

ECHEMQS
Field: yes History: yes .fil: no .dat: no

Sum of heat generation surface fluxes, Q_1 and Q_2 above.

Bond Failure Quantities

DBT Field: yes History: yes .fil: yes .dat: yes

Time when bond failure occurs.

DBS Field: yes History: yes .fil: yes .dat: yes

All components of remaining stress in the failed bond.

DBSF Field: yes History: yes .fil: yes .dat: yes

Fraction of stress that remains at bond failure.

BDSTAT Field: yes History: yes .fil: yes .dat: yes

Bond state (varies from 1.0 to 0.0).

CSDMG Field: yes History: yes .fil: yes .dat: yes

Damage variable.

OPENBC Field: yes History: yes .fil: yes .dat: yes

Relative displacement behind crack when fracture criterion is met.

CRSTS Field: yes History: yes .fil: yes .dat: yes

All components of critical stress at failure.

ENRRT Field: yes History: yes .fil: yes .dat: yes

All components of strain energy release rate.

EFENRRTR Field: yes History: yes .fil: yes .dat: yes

Effective energy release rate ratio.

CYCLE Field: yes History: yes .fil: no .dat: no

Number of cycles to debond.

CRKLENGTH Field: yes History: yes .fil: no .dat: no

Accumulated crack length at each secondary surface node, measured starting from the unbonded nodes immediately behind the initial crack front.

Cavity Radiation Variables

The output variables listed below are available in Abaqus/Standard.

RADFL Field: yes History: yes .fil: yes .dat: yes

Radiation flux per unit area.

RADFLA Field: yes History: yes .fil: yes .dat: yes

Radiation flux over the facet.

RADTL Field: yes History: yes .fil: yes .dat: yes

Time integrated radiation per unit area.

RADTLA Field: yes History: yes .fil: yes .dat: yes

Time integrated radiation over the facet.

VFTOT Field: yes History: yes .fil: yes .dat: yes

Total view factor for the facet (sum of view factor values in the row of view factor matrix corresponding to the facet).

FTEMP Field: yes History: yes .fil: yes .dat: yes

Facet temperature.

References:

- *Requesting Surface Variable Output*
- *Cavity Radiation Output in Abaqus/Standard*

Section Variables

The output variables listed below are available in Abaqus/Standard.

References:

- *Section Output from Abaqus/Standard*
- *Integrated Output*
- *Integrated Output Section Definition*

All Analysis Types

SOAREA Field: no History: yes .fil: yes .dat: yes

Area of the defined section.

Stress/Displacement Analysis

SOF Field: no History: yes .fil: yes .dat: yes

Total force in the section.

SOM Field: no History: yes .fil: yes .dat: yes

Total moment in the section.

SOCF Field: no History: yes .fil: yes .dat: yes

Center of the total force in the section.

Heat Transfer Analysis

SOH Field: no History: yes .fil: yes .dat: yes

Total heat flux associated with the section.

Electrical Analysis

SOE Field: no History: yes .fil: yes .dat: yes

Total current associated with the section.

Mass Diffusion Analysis

SOD Field: no History: yes .fil: yes .dat: yes

Total mass flow associated with the section.

Coupled Pore Fluid Diffusion-Stress Analysis

SOP

Field: no History: yes .fil: yes .dat: yes

Total pore fluid volume flux associated with the section.

Coupled Thermal-Electrochemical-Mechanical Analysis

ECHEMQSA1

Field: no History: yes .fil: no .dat: yes

Total surface heat generation power of Butler-Volmer electric current, Q_1 (area integral of ECHEMQS1).

ECHEMQSA2

Field: no History: yes .fil: no .dat: yes

Total surface entropic heat generation power, Q_2 (area integral of ECHEMQS2).

ECHEMQSA

Field: no History: yes .fil: no .dat: yes

Sum of total surface heat generation powers, Q_1 and Q_2 above.

Whole and Partial Model Variables

The output variables listed below are available in Abaqus/Standard.

References:

Adaptive Mesh Domains

The following variable is available only for adaptive domains (see *Defining ALE Adaptive Mesh Domains in Abaqus/Standard*).

VOLC Field: no History: yes .fil: yes .dat: yes

Change in area or change in volume of an element set solely due to adaptive meshing.

Equivalent Rigid Body Motion Variables

You can request equivalent rigid body motion whole element set variable output to the data, results, or output database file (see *Element Output* and *Writing Element Output to the Output Database*). The variables listed are available only for implicit dynamic analyses using direct integration except where indicated.

XC Field: no History: yes .fil: yes .dat: yes

Current coordinates of the center of mass for the entire set or the entire model. Not available for eigenfrequency extraction, eigenvalue buckling prediction, complex eigenfrequency extraction, or linear dynamics procedures. Available also for static analyses but only from the output database.

XC*n* Field: no History: yes .fil: no .dat: yes

Coordinate *n* of the center of mass for the entire set or the entire model ($n = 1, 2, 3$).

UC Field: no History: yes .fil: yes .dat: yes

Current displacement of the center of mass for the entire set or the entire model. Available also for static analyses but only from the output database.

UC*n* Field: no History: yes .fil: no .dat: yes

Displacement component *n* of the center of mass for the entire set or the entire model ($n = 1, 2, 3$).

URC*n* Field: no History: yes .fil: no .dat: yes

Rotation component *n* of the center of mass for the entire set or the entire model ($n = 1, 2, 3$).

VC Field: no History: yes .fil: yes .dat: yes

Equivalent rigid body velocity components summed over the entire set or the entire model.

VC*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the equivalent rigid body velocity summed over the entire set or the entire model ($n = 1, 2, 3$).

VRC*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the equivalent rigid body angular velocity summed over the entire set or the entire model ($n = 1, 2, 3$).

HC Field: no History: yes .fil: yes .dat: yes

Current angular momentum about the center of mass for the entire set or the entire model.

HC*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the angular momentum about the center of mass for the entire set or the entire model ($n = 1, 2, 3$).

HO Field: no History: yes .fil: yes .dat: yes

Current angular momentum about the origin for the entire set or the entire model.

HO*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the angular momentum about the origin for the entire set or the entire model ($n = 1, 2, 3$).

RI Field: no History: yes .fil: yes .dat: yes

Current rotary inertia about the origin of the entire set or the entire model. Not available for eigenfrequency extraction, eigenvalue buckling prediction, complex eigenfrequency extraction, or linear dynamics procedures. Available also for static analyses but only from the output database.

RI*ij* Field: no History: yes .fil: no .dat: yes

ij-component of the rotary inertia about the origin of the entire set or the entire model ($i \le j \le 3$).

MASS Field: no History: yes .fil: yes .dat: yes

Current mass of the entire set or the entire model. Available also for static analyses but only from the output database.

VOL Field: no History: yes .fil: yes .dat: yes

Current volume of the entire set or the entire model. Available also for static analyses but only from the output database. (Available only for continuum and structural elements that do not use general beam or shell section definitions.)

Inertia Relief Output Variables

You can request inertia relief whole model variable output to the data or output database file (see *Element Output* and *Writing Element Output to the Output Database*). Since these variables have unique values for the entire model, the variable output is independent of the specified region. The variables listed are available only for those analyses that include inertia relief loading (see *Inertia Relief*).

IRX Field: no History: yes .fil: no .dat: yes

Current coordinates of the reference point.

IRX*n* Field: no History: yes .fil: no .dat: yes

Coordinate *n* of the reference point ($n = 1, 2, 3$).

IRA Field: no History: yes .fil: no .dat: yes

Equivalent rigid body acceleration components.

IRA*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the equivalent rigid body acceleration ($n = 1, 2, 3$).

IRAR*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the equivalent rigid body angular acceleration with respect to the reference point ($n = 1, 2, 3$).

IRF Field: no History: yes .fil: no .dat: yes

Inertia relief load corresponding to the equivalent rigid body acceleration.

IRF*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the inertia relief load corresponding to the equivalent rigid body acceleration ($n = 1, 2, 3$).

IRM*n* Field: no History: yes .fil: no .dat: yes

Component *n* of the inertia relief moment corresponding to the equivalent rigid body angular acceleration with respect to the reference point ($n = 1, 2, 3$).

IRRI Field: no History: yes .fil: no .dat: yes

Rotary inertia about the reference point.

IRRI*ij* Field: no History: yes .fil: no .dat: yes

ij-component of the rotary inertia about the reference point ($i \leq j \leq 3$).

IRMASS Field: no History: yes .fil: no .dat: yes

Whole model mass.

Mass Diffusion Analysis

You can request variable output from a mass diffusion analysis (*Mass Diffusion Analysis*) to the data, results, or output database file (see *Element Output* and *Writing Element Output to the Output Database*). If you specify an output region, the variable is calculated over the user-specified region. If you do not specify an output region, the variable is calculated as the total over the entire model.

SOL Field: no History: yes .fil: yes .dat: yes

Amount of solute in an element set, calculated as the sum of ESOL (amount of solute in each element) over all the elements in the set.

Analyses with Time-Dependent Material Behavior

CRPTIME Field: no History: yes .fil: no .dat: no

Creep time, which is equal to the total time in procedures with time-dependent material behavior (see *Rate-Dependent Plasticity: Creep and Swelling*).

Eigenvalue Extraction

The following variables are output automatically during a frequency extraction analysis (*Natural Frequency Extraction*).

EIGVAL Field: no History: automatic .fil: no .dat: automatic

Eigenvalues.

EIGFREQ Field: no History: automatic .fil: no .dat: automatic

Eigenfrequencies.

GM Field: no History: automatic .fil: no .dat: automatic

Generalized masses.

CD Field: no History: automatic .fil: no .dat: automatic

Composite damping factors.

PF*n* Field: no History: automatic .fil: no .dat: automatic

Modal participation factors 1–7 ($n = 1, 2, 3$ corresponding to displacements, $n = 4, 5, 6$ for the rotations, and $n = 7$ for acoustic pressure).

EM*n* Field: no History: automatic .fil: no .dat: automatic

Modal effective masses 1–7 ($n = 1, 2, 3$ corresponding to displacements, $n = 4, 5, 6$ for the rotations, and $n = 7$ for acoustic pressure).

Complex Eigenvalue Extraction

The following variables are output automatically during a complex frequency extraction analysis (*Complex Eigenvalue Extraction*).

EIGREAL Field: no History: automatic .fil: no .dat: automatic

Real parts of the eigenvalues.

EIGIMAG Field: no History: automatic .fil: no .dat: automatic

Imaginary parts of the eigenvalues.

EIGFREQ Field: no History: automatic .fil: no .dat: automatic

Eigenfrequencies.

DAMPRATIO Field: no History: automatic .fil: no .dat: automatic

Damping ratios.

Total Energy Output Quantities

If the following whole model variables are relevant for a particular analysis, you can request them as output to the data, results, or output database file (see *Total Energy Output* and *Total Energy Output*). If you do not specify an output region, whole model variables are calculated. When you specify an output region, the relevant energy totals are calculated over the user-specified region.

These variables are not available for eigenvalue buckling prediction, eigenfrequency extraction, or complex frequency extraction analysis. You cannot specify an output region for modal dynamic, random response, response spectrum, or steady-state dynamic analysis.

See *Energy balance* and *Energy computations in a contact analysis* for details of the energy definitions. In steady-state dynamics all energy quantities are net per-cycle values, unless otherwise noted.

ALLAE Field: no History: yes .fil: automatic .dat: automatic

"Artificial" strain energy associated with constraints used to remove singular modes (such as hourglass control), and with constraints used to make the drill rotation follow the in-plane rotation of the shell elements.

ALLCCDW Field: no History: yes .fil: no .dat: automatic

Contact constraint discontinuity work.

ALLCCEN Field: no History: yes .fil: no .dat: automatic

Contact constraint elastic energy in normal direction due to penalty constraint enforcement.

ALLCCET Field: no History: yes .fil: no .dat: automatic

Contact constraint elastic energy in tangential direction due to friction penalty constraint enforcement.

ALLCCE Field: no History: yes .fil: no .dat: automatic

The sum of ALLCCEN and ALLCCET.

ALLCCSDN Field: no History: yes .fil: no .dat: automatic

Contact constraint stabilization dissipation in normal direction.

ALLCCSDT Field: no History: yes .fil: no .dat: automatic

Contact constraint stabilization dissipation in tangential direction.

ALLCCSD Field: no History: yes .fil: no .dat: automatic

The sum of ALLCCSDN and ALLCCSDT.

ALLCD Field: no History: yes .fil: automatic .dat: automatic

Energy dissipated by creep, swelling, viscoelasticity, and energy associated with viscous regularization for cohesive elements and cohesive contact.

ALLEE Field: no History: yes .fil: automatic .dat: automatic

Electrostatic energy.

ALLFD Field: no History: yes .fil: automatic .dat: automatic

Total energy dissipated through frictional effects. (Available only for the whole model.)

ALLIE Field: no History: yes .fil: automatic .dat: automatic

Total strain energy. (ALLIE = ALLSE + ALLPD + ALLCD + ALLAE + ALLQB + ALLEE + ALLDMD.)

ALLJD Field: no History: yes .fil: automatic .dat: automatic

Electrical energy dissipated due to flow of electrical current.

ALLKE Field: no History: yes .fil: automatic .dat: automatic

Kinetic energy. In steady-state dynamic and frequency extraction analyses, this is the cyclic mean value. In frequency extraction analyses, the value of kinetic energy is normalized. Normalization is performed for each eigenmode separately, such that the kinetic and strain energies for the whole model add up to one.

ALLKEA Field: no History: yes .fil: no .dat: no

Kinetic energy amplitude. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLKEP Field: no History: yes .fil: no .dat: no

Kinetic energy peak value. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLKL Field: no History: yes .fil: automatic .dat: automatic

Loss of kinetic energy at impact. (Available only for the whole model.)

ALLPD Field: no History: yes .fil: automatic .dat: automatic

Energy dissipated by rate-independent and rate-dependent plastic deformation. For superelastic materials, this variable also includes recoverable phase-transformation energy.

ALLQB Field: no History: yes .fil: automatic .dat: automatic

Energy dissipated through quiet boundaries (infinite elements). (Available only for the whole model.)

ALLSD Field: no History: yes .fil: automatic .dat: automatic

Energy dissipated by automatic stabilization. This includes both volumetric static stabilization and automatic approach of contact pairs (the latter part included only for the whole model).

ALLSE Field: no History: yes .fil: automatic .dat: automatic

Recoverable strain energy. In steady-state dynamic and frequency extraction analyses, this is the cyclic mean value. In frequency extraction analyses, the value of strain energy is normalized. Normalization is performed for each eigenmode separately, such that the kinetic and strain energies for the whole model add up to one.

ALLSEA Field: no History: yes .fil: no .dat: no

Recoverable strain energy amplitude. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLSEP Field: no History: yes .fil: no .dat: no

Recoverable strain energy peak value. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLVD Field: no History: yes .fil: automatic .dat: automatic

Energy dissipated by viscous effects including viscous regularization (except for cohesive elements and cohesive contact), not inclusive of energy dissipated by automatic stabilization and viscoelasticity. If this variable is requested for the whole model in mode-based steady-state dynamic analyses, it includes the energy loss due to the material, global, and modal damping. If this variable is requested for a part of the model in mode-based steady-state dynamic analyses, the energy loss due to the modal damping is not included. In direct-solution steady-state dynamic analyses this variable includes the energy loss due to the material and global damping.

ALLVDE Field: no History: yes .fil: no .dat: automatic

Energy dissipated by viscous effects due to the material damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLVDG Field: no History: yes .fil: no .dat: automatic

Energy dissipated by viscous effects due to the global damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLVDM Field: no History: yes .fil: no .dat: automatic

Energy dissipated by viscous effects due to the modal damping. This variable is available only for the whole model in the mode-based steady-state dynamic analyses.

ALLHD Field: no History: yes .fil: no .dat: automatic

Energy dissipated due to the structural damping. If this variable is requested for the whole model in mode-based steady-state dynamic analyses, it includes energy loss due to the material, global, and modal damping. If this variable is requested for a part of the model in mode-based steady-state dynamic analyses, energy loss due to the modal damping is not included. In direct-solution steady-state dynamic analyses this variable includes the energy loss due to the material and global damping.

ALLHDE Field: no History: yes .fil: no .dat: automatic

Energy dissipated due to the material structural damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLHDG Field: no History: yes .fil: no .dat: automatic

Energy dissipated due to the global structural damping. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLHDM Field: no History: yes .fil: no .dat: automatic

Energy dissipated due to the modal structural damping. This variable is available only for the whole model in mode-based steady-state dynamic analyses.

ALLDMD Field: no History: yes .fil: automatic .dat: automatic

Energy dissipated by damage.

ALLWK Field: no History: yes .fil: automatic .dat: automatic

External work. (Available only for the whole model.)

ETOTAL Field: no History: yes .fil: automatic .dat: automatic

Total energy balance (available only for the whole model). (ETOTAL = ALLKE + ALLIE + ALLVD + ALLSD + ALLKL + ALLFD + ALLJD + ALLCCE − ALLWK − ALLCCDW.)

EFLOW Field: no History: yes .fil: no .dat: automatic

Energy flow from a given portion of the model through the given boundary. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

PFLOW Field: no History: yes .fil: no .dat: automatic

Power flow from a given portion of the model through the given boundary. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

RADEN Field: no History: yes .fil: no .dat: automatic

Radiated energy from/into a given acoustic cavity through the given boundary. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

RADPOW Field: no History: yes .fil: no .dat: automatic

Radiated power from/into a given acoustic cavity through the given boundary. This variable is available only in mode-based and direct-solution steady-state dynamic analyses.

ALLHUMDFLUX Field: no History: yes .fil: no .dat: automatic

All heat energy due to the nonuniform moving flux prescribed inside user subroutine *UMDFLUX*. This variable is available only in pure heat transfer analyses. (Available only for the whole model.)

ALLUSER Field: no History: yes .fil: no .dat: no

User-defined quantity that can be set only in user subroutines for pure heat transfer analyses. Its value is set through calls to the utility routine `SETALLUSER`.

ALLERPWR Field: no History: yes .fil: no .dat: no

Equivalent radiated power emitted by a panel. This variable is available only in steady-state dynamic analyses.

Total Amount Output Quantities

The following whole model variables are relevant only for electrochemical analyses. You can request them as history output to the output database file. If you do not specify an output region as an element set in the element output, whole model variables are calculated. When you specify an output region, the relevant amount totals are calculated over the user-specified region. Amount is measured in moles. While the output variables presented here are in terms of lithium metal, in general, you can replace lithium with another chemical species.

AMOUNT Field: no History: yes .fil: no .dat: no

Total amount of lithium in the solid particles and the electrolyte (sum of AMOUNTE and AMOUNTS).

AMOUNTE Field: no History: yes .fil: no .dat: no

Amount of lithium in the electrolyte.

AMOUNTS Field: no History: yes .fil: no .dat: no

Amount of lithium in the solid particles.

Total Volume Heat Generation Power Output Quantities

The following whole model variables are relevant only for electrochemical analyses. You can request them as history output to the output database file. If you do not specify an output region as an element set in the element output, whole model variables are calculated. When you specify an output region, the relevant power totals are calculated over the user-specified region.

ECHEMQV1 Field: no History: yes .fil: no .dat: no

Total volume heat generation power of Butler-Volmer electric current, Q_1 (volume integral of ECHEMQ1).

ECHEMQV2 Field: no History: yes .fil: no .dat: no

Total volume entropic heat generation power, Q_2 (volume integral of ECHEMQ2).

ECHEMQV3 Field: no History: yes .fil: no .dat: no

Total volume heat generation power of electronic electric current, Q_3 (volume integral of ECHEMQ3).

ECHEMQV4 Field: no History: yes .fil: no .dat: no

Total volume heat generation power of ionic electric current, Q_4 (volume integral of ECHEMQ4).

ECHEMQV5 Field: no History: yes .fil: no .dat: no

Total volume heat generation power of ionic diffusive flux, Q_5 (volume integral of ECHEMQ5).

Total State of Charge

The following whole model variable is relevant only for electrochemical analyses. You can request it as history output to the output database file. If you do not specify an output region as an element set in the element output, the whole model variable is calculated. When you specify an output region, the relevant state of charge total is calculated over the user-specified region.

SOC Field: no History: yes .fil: no .dat: no

Current state of charge in the specified region of the model.

Solution-Dependent Amplitude Variables

The output variables listed below are available in Abaqus/Standard.

LPF Field: no History: automatic .fil: automatic .dat: no

Load proportionality factor in a static Riks analysis.

AMPCU Field: no History: automatic .fil: automatic .dat: no

Current value of the solution-dependent amplitude.

RATIO Field: no History: automatic .fil: automatic .dat: no

Current maximum ratio of creep strain rate and target creep strain rate.

References:

Structural Optimization Variables

The output variables listed below are available in Abaqus/Standard.

Topology Optimization

The following variable is output automatically during topology optimization (see *Topology Optimization*).

MAT_PROP_NORMALIZED Field: automatic History: no .fil: no .dat: no

Element-based normalized material value.

Shape Optimization

The following variables are output automatically during shape optimization (see *Shape Optimization*).

CTRL_INPUT Field: automatic History: no .fil: no .dat: no

The value of the objective function at each node.

DISP_NORMAL_VAL Field: automatic History: no .fil: no .dat: no

The value of the bead optimization displacement along the node normal vector.

DISP_OPT_VAL Field: automatic History: no .fil: no .dat: no

The value of the shape optimization displacement.

DISP_OPT Field: automatic History: no .fil: no .dat: no

A vector representing the shape optimization displacement.

Sizing Optimization

The following variables are output automatically during sizing optimization (see *Sizing Optimization*).

THICKNESS Field: automatic History: no .fil: no .dat: no

The value of the shell thickness.

DELTA_THICKNESS Field: automatic History: no .fil: no .dat: no

The change in shell thickness.

Bead Optimization

The following variables are output automatically during bead optimization (see *Bead Optimization*).

DISP_NORMAL_VAL Field: automatic History: no .fil: no .dat: no

The value of the bead optimization displacement along the node normal vector.

DISP_OPT_VAL Field: automatic History: no .fil: no .dat: no

The value of the bead optimization displacement.

DISP_OPT Field: automatic History: no .fil: no .dat: no

A vector representing the bead optimization displacement.

Using Abaqus/Explicit Output Variable Identifiers

In this section:

Abaqus/Explicit Output Variable Identifiers

Products: Abaqus/Explicit

References:

- *About Output*
- *Output to the Data and Results Files*
- *Output to the Output Database*
- *Loads*
- *Abaqus/Explicit User Subroutines*

Overview

Except for the information in the status file, results can be obtained from Abaqus/Explicit only by postprocessing.

The tables in the following sections list all of the output variables that are available in Abaqus/Explicit. These output variables can be requested as either field- or history-type output to the output database (`.odb`) file (see *Output to the Output Database*) or for output to the results (`.fil`) file (see *Output to the Data and Results Files*). In general, output variables that can be requested as field- or history-type output to an output database in ODB format can also be requested as output in SIM format (see *The Output Database*). When the output variables are requested for output to the results file, Abaqus/Explicit first writes these variables to the selected results (`.sel`) file and then converts the selected results file to the results file after the analysis completes.

Notation Used in the Output Variable Descriptions

The entries “Field”, “History”, and “.fil” in the output variable's description indicate the availability of the output variable. “Field” refers to a field-type output selection to the output database, “History” refers to a history-type output selection to the output database, and “.fil” refers to a results file output selection. The output variable can be written to the respective file if the word “yes” appears after the category name; “no” means that the variable is not available to that file.

Direction Definitions

The direction definitions depend on the variable type.

Direction Definitions for Element Variables

For components of stress, strain, and similar material variables, 1, 2, and 3 refer to the directions in an orthogonal coordinate system. These are global directions for solid elements, surface directions for shell and membrane elements, and axial and transverse directions for beam and pipe elements. However, if a local orientation (*Orientations*) is associated with the elements for which output is being requested, 1, 2, and 3 are local directions.

Direction Definitions for Nodal Variables

For nodal variables, 1, 2, and 3 refer to the global directions (1=*X*, 2=*Y*, 3=*Z* except for axisymmetric elements, in which case 1=*R*, 2=*Z*). Even if a local coordinate system has been defined at a node (*Transformed Coordinate Systems*), the data in the results file and the selected results file are still output in the global directions.

If nodal field output is requested for a node that has a local coordinate system defined, a quaternion representing the rotation from the global directions is written to the output database.Abaqus/CAE automatically uses this quaternion to transform the nodal results into the local directions. Nodal history data written to the output database are always stored in the global directions.

Direction Definitions for Integrated Variables

For components of total force, total moment, and similar variables obtained through integration over a surface, the directions 1, 2, and 3 refer to directions in an orthogonal coordinate system. A fixed global coordinate system is used if the surface is specified directly for the integrated output request. If the surface is identified by an integrated output section definition (see *Integrated Output Section Definition*) that is associated with the integrated output request, a local coordinate system in the initial configuration can be specified and can translate or rotate with the deformation.

Distributed Load Output and User Subroutines

Output can be requested for many of the distributed loads discussed in *Loads*. However, contributions to these loads defined through user subroutines (see *Abaqus/Explicit User Subroutines*) are not displayed.

Principal Value Output

Output of the principal values can be requested for stresses, logarithmic strains, and nominal strains. Either all principal values or the minimum, intermediate, or maximum values can be obtained. All principal values of tensor *ABC* are obtained with the request *ABC*P, and the minimum, intermediate, and maximum principal values are obtained with the requests *ABC*P1, *ABC*P2, and *ABC*P3, respectively. For three-dimensional, plane strain, and axisymmetric elements all three principal values are obtained. For plane stress, membrane, and shell elements only the in-plane principal values are obtained for history-type output, and the out-of-plane principal value cannot be requested. For field-type output, Abaqus/CAE always assumes the out-of-plane principal value as zero, including when computing the **Max. Principal**, **Mid. Principal**, and **Min. Principle** values. Principal values cannot be obtained for beam, pipe, and truss elements, and principal values of plastic strains cannot be requested.

If a principal value or an invariant is requested for field-type output, the output request is replaced with an output request for the components of the corresponding tensor. Abaqus/CAE calculates all principal values and invariants from these components. If a principal value is desired as history-type output, it must be requested explicitly since Abaqus/CAE does no calculations on history data.

Tensor Output

Tensor variables that are written to the output database as field-type output are written as components in either the default directions defined by the convention given in *Orientations* (global directions for solid elements, surface directions for shell and membrane elements, and axial and transverse directions for beam and pipe elements), or the user-defined local system. Abaqus/CAE calculates all principal values and invariants from these components. See *Writing field output data*, for a description of the different types of tensor variables.

The components for tensor variables are written to the output database in single precision. Therefore, a small amount of precision roundoff error can occur when calculating the variables' principal values. Such roundoff error may be observed, for example, when analytically zero values are calculated as relatively small yet nonzero values.

Requesting Output of Components

Individual components of variables can be requested as history-type output in the output database for *X–Y* plotting in Abaqus/CAE. Individual component requests are not available for field-type output. If a particular component is required for contouring in Abaqus/CAE, request field output of the generic variable (for example, S for stress). Output for individual components of this field output can be requested within the Visualization module of Abaqus/CAE.

Element Integration Point Variables

The output variables listed below are available in Abaqus/Explicit.

References:

- *Element Output*
- *Writing Element Output to the Output Database*

Tensors and Invariants

S
Field: yes History: yes .fil: yes

All stress components.

MISESMAX
Field: yes History: no .fil: no

Maximum Mises stress among all the section points. For a shell element, it represents the maximum Mises value among all the section points in the layer, for a beam or pipe element it is the maximum Mises stress among all the section points in the cross-section, and for a solid element it represents the Mises stress at the integration points.

S*ij*
Field: no History: yes .fil: no

ij-component of stress ($i \le j \le 3$).

SP
Field: yes History: yes .fil: yes

All principal stress components.

SP*n*
Field: no History: yes .fil: no

Minimum, intermediate, and maximum principal stress components (SP1≤SP2≤SP3).

E
Field: yes History: yes .fil: yes

All infinitesimal strain components for geometrically linear analysis.

E*ij*
Field: no History: yes .fil: no

ij-component of infinitesimal strain ($i \le j \le 3$).

LE
Field: yes History: yes .fil: yes

All logarithmic strain components.

LE*ij*
Field: no History: yes .fil: no

ij-component of logarithmic strain ($i \le j \le 3$).

LEP
Field: yes History: yes .fil: yes

All principal logarithmic strain components.

LEP*n*
Field: no History: yes .fil: no

Minimum, intermediate, and maximum principal logarithmic strain components (LEP1≤LEP2≤LEP3).

THE Field: yes History: yes .fil: yes

All thermal strain components.

THE*ij* Field: no History: yes .fil: no

ij-component of thermal strain ($i \leq j \leq 3$).

THEP Field: yes History: yes .fil: yes

All principal thermal strains.

THEP*n* Field: no History: yes .fil: no

Minimum, intermediate, and maximum principal thermal strains (THEP1≤THEP2≤THEP3).

ER Field: yes History: yes .fil: yes

All logarithmic strain rate components.

ER*ij* Field: no History: yes .fil: no

ij-component of logarithmic strain rate($i \leq j \leq 3$).

ERP Field: yes History: yes .fil: yes

All principal logarithmic strain rate components.

ERP*n* Field: no History: yes .fil: no

Minimum, intermediate, and maximum principal strain rate components (ERP1≤ERP2≤ERP3).

NE Field: yes History: yes .fil: yes

All nominal strain components.

NE*ij* Field: no History: yes .fil: no

ij-component of nominal strain ($i \leq j \leq 3$).

NEP Field: yes History: yes .fil: yes

All principal nominal strain components.

NEP*n* Field: no History: yes .fil: no

Minimum, intermediate, and maximum principal nominal strain components (NEP1≤NEP2≤NEP3).

PE Field: yes History: yes .fil: yes

All plastic strain components.

PE*ij* Field: no History: yes .fil: no

ij-component of plastic strain ($i \leq j \leq 3$).

PEP Field: yes History: yes .fil: no

All principal plastic strains.

PEP*n* Field: no History: yes .fil: no

Minimum, intermediate, and maximum principal plastic strains.

CE Field: yes History: yes .fil: no

All creep strain components.

CE*ij* Field: no History: yes .fil: no

ij-component of creep strain ($i \leq j \leq 3$).

ERV Field: yes History: yes .fil: yes

Volumetric strain rate.

MISES Field: yes History: yes .fil: yes

Mises equivalent stress, defined as $q = \sqrt{\frac{3}{2}\mathbf{S} : \mathbf{S}}$, where $\mathbf{S}$ is the deviatoric stress tensor, defined as $\mathbf{S} = \boldsymbol{\sigma} + p\mathbf{I}$, where $\boldsymbol{\sigma}$ is the stress and $p = -\frac{1}{3}\,\mathrm{trace}\,(\boldsymbol{\sigma})$ is the equivalent pressure stress.

PRESS Field: yes History: yes .fil: yes

Equivalent pressure stress, $p = -\frac{1}{3}\,\mathrm{trace}\,(\boldsymbol{\sigma})$.

TRIAX Field: yes History: yes .fil: no

Stress triaxiality, $\eta = -p/q$.

YIELDS Field: yes History: yes .fil: no

Yield stress, σ^0, available for Mises, Tresca, Hosford, Johnson-Cook, Hill, Barlat, and extended Drucker-Prager plasticity material models.

YIELDCRT Field: yes History: yes .fil: no

Yield criterion, available for extended Drucker-Prager plasticity material models.

YIELDPOT Field: yes History: yes .fil: no

Yield potential, $f(\boldsymbol{\sigma})$, available for Tresca, Hosford, Hill, and Barlat plasticity material models.

MASSADJUST Field: yes History: no .fil: no

Adjusted or redistributed mass in each element that is included in the element sets used with mass adjustment. This output is available only in the first output frame of the first analysis step.

ALPHA Field: yes History: yes .fil: yes

All total kinematic hardening shift tensor components.

ALPHA*ij* Field: no History: yes .fil: no

ij-component of the total shift tensor ($i \leq j \leq 3$).

ALPHAP Field: yes History: yes .fil: yes

All principal values of the total shift tensor.

ALPHAP*n* Field: no History: yes .fil: no

Minimum, intermediate, and maximum principal values of the total shift tensor (ALPHAP1≤ALPHAP2≤ALPHAP3).

SNET*k* Field: yes History: yes .fil: no

All stress components in the k^{th} network ($0 \leq k \leq 10$). Available only for the parallel rheological framework.

SNET*k_ij* Field: no History: yes .fil: no

ij-component of stress in the k^{th} network ($i \leq j \leq 3$ and $0 \leq k \leq 10$). Available only for the parallel rheological framework.

PEEQ Field: yes History: yes .fil: yes

Equivalent plastic strain.

For porous metal plasticity PEEQ is the equivalent plastic strain in the matrix material defined as $\int \frac{\sigma : d\varepsilon^{pl}}{(1-f)\sigma_y}$.

For cap plasticity PEEQ gives p_b (the cap position).

For crushable foam plasticity with volumetric hardening PEEQ gives the volumetric compacting plastic strain defined as $-\varepsilon^{pl}_{\text{vol}}$.

For crushable foam plasticity with isotropic hardening PEEQ gives the equivalent plastic strain defined as $\int \frac{\sigma : d\varepsilon^{pl}}{\sigma_c}$, where σ_c is the uniaxial compression yield stress.

PEEQT Field: yes History: yes .fil: no

Equivalent plastic strain in uniaxial tension for cast iron, Mohr-Coulomb tension cutoff, and concrete damaged plasticity, which is defined as $\int \dot{\bar{\varepsilon}}^{pl}_t \, dt$.

PEEQR Field: yes History: yes .fil: no

Equivalent plastic strain rate. Available only for rate-dependent viscoplastic models.

PEEQMAX Field: yes History: no .fil: no

Maximum equivalent plastic strain, PEEQ, among all of the section points. For a shell element it represents the maximum PEEQ value among all the section points in the layer, for a beam or a pipe element it is the maximum PEEQ among all the section points in the cross-section, and for a solid element it represents the PEEQ at the integration points.

CEEQ Field: yes History: yes .fil: no

Equivalent creep strain.

DMICRTMAX Field: yes History: no .fil: no

Maximum damage initiation criteria value among all section points, all damage initiation criteria, and all time increments up to the current time.

This output variable generates three output quantities as follows:

DMICRTMAXVAL outputs the maximum damage initiation value.

DMICRTPOS outputs the section point in the layer in which the maximum damage initiation value occurred. For solid elements, the output value is one.

DMICRTTYPE outputs a value that represents the damage initiation criteria type that reached the maximum value in the element as follows:

For elements that have failure with progressive damage: 1-DUCTCRT, 2-SHRCRT, 3-JCCRT, 4-FLDCRT, 5-MSFLDCRT, 6-FLSDCRT, and 7-MKCRT.

For elements that have fiber-reinforced material damage: 11-HSNFTCRT, 12-HSNFCCRT, 13-HSNMTCRT, and 14-HSNMCCRT.

For cohesive elements with traction-separation behavior: 21-MAXSCRT, 22-MAXECRT, 23-QUADSCRT, and 24-QUADECRT.

STATUSMP Field: yes History: no .fil: no

Status of each material point in the element (1.0 if the material point is active, 0.0 if it is not active).

LODE Field: yes History: yes .fil: no

Lode angle term, $\xi = \cos(3\Theta)$, where Θ is the Lode angle.

ORITENS Field: yes History: no .fil: no

All orientation tensor components. Available only for elements with multiscale material or linear orthotropic elastic material with fiber dispersion and only if the orientation tensor is specified with a distribution (*Distribution Definition*). Only written to the output database (`.odb`) file for the original field output frame at zero time.

Geometric Quantities

COORD Field: yes History: yes .fil: no

Coordinates of the integration point. These are the current coordinates if the large-displacement formulation is being used.

IVOL Field: yes History: yes .fil: no

Integration point volume. (Available only for continuum elements.)

LOCALDIR*n* Field: automatic History: no .fil: no

Direction cosines of the local material directions for an anisotropic hyperelastic material model, or yarn direction cosines for a fabric material model. This variable is output automatically if any other element field output is requested for anisotropic hyperelastic or fabric material (see *Output* and *Output*).

Additional Element Stresses

TSHR Field: yes History: yes .fil: yes

All transverse shear stress components for three-dimensional conventional shell elements.

TSHR13 Field: no History: yes .fil: no

13-component of transverse shear stress.

TSHR23 Field: no History: yes .fil: no

23-component of transverse shear stress.

Energy Densities

ENER Field: yes History: yes .fil: yes

All energy densities.

SENER Field: yes History: yes .fil: no

Elastic strain energy density, per unit volume.

PENER Field: yes History: yes .fil: no

Energy dissipated by rate-independent and rate-dependent plasticity, per unit volume. For superelastic materials, this variable also includes recoverable phase-transformation energy.

CENER Field: yes History: yes .fil: no

Energy dissipated by viscoelasticity, per unit volume. (Not supported for hyperelastic and hyperfoam material models with linear viscoelasticity.)

VENER Field: yes History: yes .fil: no

Energy dissipated by viscous effects, per unit volume.

DMENER Field: yes History: yes .fil: no

Energy dissipated by damage, per unit volume.

State and Field Variables

SDV Field: yes History: yes .fil: yes

Solution-dependent state variables.

SDV*n* Field: yes History: yes .fil: no

Solution-dependent state variable *n*.

TEMP Field: yes History: yes .fil: yes

Temperature.

DENSITY Field: yes History: yes .fil: no

Material density.

FV Field: yes History: yes .fil: no

Field variables.

FV*n* Field: no History: yes .fil: no

Field variable *n*.

Composite Failure Measures

CFAILURE
Field: yes History: no .fil: no

All failure measure components.

MSTRS
Field: no History: no .fil: no

Maximum stress theory failure measure.

TSAIH
Field: no History: no .fil: no

Tsai-Hill theory failure measure.

TSAIW
Field: no History: no .fil: no

Tsai-Wu theory failure measure.

AZZIT
Field: no History: no .fil: no

Azzi-Tsai-Hill theory failure measure.

MSTRN
Field: no History: no .fil: no

Maximum strain theory failure measure.

Additional Plasticity Quantities

PEQC
Field: yes History: yes .fil: yes

All equivalent plastic strains, when the model has more than one yield/failure surface.

PEQC*n*
Field: no History: yes .fil: no

*n*th equivalent plastic strain ($n = 1, 2, 3, 4$).

For cap plasticity: PEQC provides equivalent plastic strains for all three possible yield/failure surfaces (Drucker-Prager failure surface - PEQC1, cap surface - PEQC2, and transition surface - PEQC3) and the total volumetric plastic strain (PEQC4). All identifiers also provide a yes/no flag (1/0 in the output database), telling whether the yield surface is currently active or not (AC YIELD: "actively yielding").

When PEQC is requested as output to the output database, the active yield flags for each component are named AC YIELD1, AC YIELD2, etc.

RD
Field: yes History: yes .fil: yes

Relative density (cap plasticity).

VVF
Field: yes History: yes .fil: yes

Void volume fraction (cap plasticity).

Porous Metal Plasticity Quantities

RD
Field: yes History: yes .fil: yes

Relative density (porous metal plasticity).

VVF Field: yes History: yes .fil: yes

Void volume fraction (porous metal plasticity).

VVFG Field: yes History: yes .fil: yes

Void volume fraction due to growth (porous metal plasticity).

VVFN Field: yes History: yes .fil: yes

Void volume fraction due to nucleation (porous metal plasticity).

Concrete Damaged Plasticity

DAMAGEC Field: yes History: yes .fil: no

Compressive damage variable, d_c.

DAMAGET Field: yes History: yes .fil: no

Tensile damage variable, d_t.

SDEG Field: yes History: yes .fil: no

Scalar stiffness degradation variable, d.

PEEQ Field: yes History: yes .fil: no

Equivalent plastic strain in uniaxial compression, which is defined as $\int \dot{\tilde{\varepsilon}}_c^{pl} dt$.

PEEQR Field: yes History: yes .fil: no

Equivalent plastic strain rate.

Superelastic Material Quantities

TE Field: yes History: yes .fil: no

Transformation strain tensor.

TE*ij* Field: no History: yes .fil: no

ij-component of transformation strain ($i \leq j \leq 3$).

TEEQ Field: yes History: yes .fil: no

Equivalent transformation strain.

TEVOL Field: yes History: yes .fil: no

Volumetric transformation strain.

MVF Field: yes History: yes .fil: no

Fraction of martensite.

SEQUT Field: yes History: yes .fil: no

Equivalent uniaxial tensile stress.

EEQUT Field: yes History: yes .fil: no

Equivalent uniaxial tensile total strain.

EE Field: yes History: yes .fil: no

Elastic strain tensor.

EE*ij* Field: yes History: yes .fil: no

ij-component of elastic strain ($i \leq j \leq 3$).

Cracking Model Quantities

CKE Field: yes History: no .fil: yes

All cracking strain components.

CKE*ij* Field: no History: no .fil: no

ij-component of cracking strain.

CKLE Field: yes History: no .fil: yes

All cracking strain components in local crack axes.

CKLE*ij* Field: no History: no .fil: no

ij-component of cracking strain in local crack axes.

CKEMAG Field: yes History: no .fil: yes

Cracking strain magnitude, defined as $\sqrt{\left(e_{nn}^{ck}\right)^2 + \left(e_{tt}^{ck}\right)^2 + \left(e_{ss}^{ck}\right)^2}$.

CKLS Field: yes History: no .fil: yes

All stress components in local crack axes.

CKLS*ij* Field: no History: no .fil: no

ij-component of stress in local crack axes.

CRACK Field: no History: no .fil: yes

Crack orientations.

CKSTAT Field: yes History: no .fil: yes

Crack status of each crack. CKSTAT can have the following values for each crack: 0.0=uncracked, 1.0=closed crack, 2.0=actively cracking, 3.0=crack closing/reopening.

Failure with Progressive Damage

DMICRT Field: yes History: yes .fil: no

All active components of the damage initiation criteria.

DUCTCRT Field: no History: yes .fil: no

Ductile damage initiation criterion.

JCCRT Field: no History: yes .fil: no

Johnson-Cook damage initiation criterion.

HCCRT Field: no History: yes .fil: no

Hosford-Coulomb damage initiation criterion.

SHRCRT Field: no History: yes .fil: no

Shear damage initiation criterion.

FLDCRT Field: no History: yes .fil: no

Forming limit diagram (FLD) damage initiation criterion.

FLSDCRT Field: no History: yes .fil: no

Forming limit stress diagram (FLSD) damage initiation criterion.

MSFLDCRT Field: no History: yes .fil: no

Müschenborn-Sonne forming limit stress diagram (MSFLD) damage initiation criterion.

MKCRT Field: no History: yes .fil: no

Marciniak-Kuczynski (M-K) damage initiation criterion.

SDEG Field: yes History: yes .fil: no

Overall scalar stiffness degradation.

ERPRATIO Field: yes History: yes .fil: no

Ratio of principal strain rates, α, used for the MSFLD damage initiation criterion.

SHRRATIO Field: yes History: yes .fil: no

Shear stress ratio, $\theta_s = (q + k_s p)\,/\tau_{\mathrm{max}}$, used for the shear damage initiation criterion.

Fiber-Reinforced Materials Damage

DMICRT Field: yes History: yes .fil: no

All active components of the damage initiation criteria.

HSNFTCRT Field: no History: yes .fil: no

Hashin's fiber tensile damage initiation criterion.

HSNFCCRT Field: no History: yes .fil: no

Hashin's fiber compressive damage initiation criterion.

HSNMTCRT Field: no History: yes .fil: no

Hashin's matrix tensile damage initiation criterion.

HSNMCCRT Field: no History: yes .fil: no

Hashin's matrix compressive damage initiation criterion.

PLF1TCRT Field: no History: yes .fil: no

Fiber tensile damage initiation criterion in the local 1-direction for the ply fabric model.

PLF1CCRT Field: no History: yes .fil: no

Fiber compressive damage initiation criterion in the local 1-direction for the ply fabric model.

PLF2TCRT Field: no History: yes .fil: no

Fiber tensile damage initiation criterion in the local 2-direction for the ply fabric model.

PLF2CCRT Field: no History: yes .fil: no

Fiber compressive damage initiation criterion in the local 2-direction for the ply fabric model.

PLSHRCRT Field: no History: yes .fil: no

Matrix shear damage initiation criterion for the ply fabric model.

DAMAGEFT Field: yes History: yes .fil: no

Fiber tensile damage variable for the Hashin model.

DAMAGEFC Field: yes History: yes .fil: no

Fiber compressive damage variable for the Hashin model.

DAMAGEMT Field: yes History: yes .fil: no

Matrix tensile damage variable for the Hashin model.

DAMAGEMC Field: yes History: yes .fil: no

Matrix compressive damage variable for the Hashin model.

DAMAGESHR Field: yes History: yes .fil: no

Shear damage variable for the Hashin model and for the ply fabric model.

DAMAGEF1T Field: yes History: yes .fil: no

Fiber tensile damage variable in the local 1-direction for the ply fabric model.

DAMAGEF1C Field: yes History: yes .fil: no

Fiber compressive damage variable in the local 1-direction for the ply fabric model.

DAMAGEF2T Field: yes History: yes .fil: no

Fiber tensile damage variable in the local 2-direction for the ply fabric model.

DAMAGEF2C Field: yes History: yes .fil: no

Fiber compressive damage variable in the local 2-direction for the ply fabric model.

General Strain/Stress-Based Damage

DMICRT Field: yes History: yes .fil: no

All active components of the damage initiation criteria.

MSTRESSCRT Field: no History: yes .fil: no

Maximum stress-based damage initiation criterion.

MSTRAINCRT Field: no History: yes .fil: no

Maximum stress-based damage initiation criterion.

TSAIWUCRT Field: no History: yes .fil: no

Tsai-Wu stress-based damage initiation criterion.

TSAIWUECRT Field: no History: yes .fil: no

Tsai-Wu strain-based damage initiation criterion.

DMIFI Field: yes History: yes .fil: no

All active components of the damage initiation failure indices.

MSTRESSFI Field: no History: yes .fil: no

Maximum stress-based damage initiation failure index.

MSTRAINFI Field: no History: yes .fil: no

Maximum stress-based damage initiation criterion.

TSAIWUFI Field: no History: yes .fil: no

Tsai-Wu stress-based damage initiation failure index.

TSAIWUEFI Field: no History: yes .fil: no

Tsai-Wu strain-based damage initiation failure index.

Fabric Material

Output variable LOCALDIR (described above) is output automatically for fabric materials.

SFABRIC Field: yes History: yes .fil: no

All fabric stress components.

EFABRIC Field: yes History: yes .fil: no

All fabric strain components.

SFABRIC*ij* Field: no History: yes .fil: no

ij-component of fabric stress ($i \leq j \leq 3$).

EFABRIC*ij* Field: no History: yes .fil: no

ij-component of fabric strain ($i \leq j \leq 3$).

Equation of State

BURNF Field: yes History: yes .fil: no

Burn fraction of the ignition and growth material.

DBURNF Field: yes History: yes .fil: no

Reaction rate of the ignition and growth material.

RHOE Field: yes History: yes .fil: no

Density of the unreacted explosive in the ignition and growth material.

RHOP Field: yes History: yes .fil: no

Density of the reacted gas product in the ignition and growth material.

PALPH Field: yes History: yes .fil: no

Distension, α, of the $P-\alpha$ porous material.

PALPHMIN Field: yes History: yes .fil: no

Minimum value, α_{min}, of the distension attained during plastic compaction of the $P-\alpha$ porous material.

Rebar Quantities

RBFOR Field: yes History: yes .fil: yes

Force in rebar.

RBANG Field: yes History: yes .fil: yes

Angle, in degrees, between rebar and the user-specified isoparametric direction. Available only for shell and membrane elements.

RBROT Field: yes History: yes .fil: yes

Change in angle, in degrees, between rebar and the user-specified isoparametric direction. Available only for shell and membrane elements.

Integration Point Coordinates

COORD Field: yes History: yes .fil: no

Coordinates of element integration point.

Coupled Thermal-Stress Elements

HFL Field: yes History: yes .fil: yes

Current magnitude and components of the heat flux per unit area vector.

HFLM Field: no History: yes .fil: no

Current magnitude of the heat flux per unit area vector.

HFL*n* Field: no History: yes .fil: no

Component *n* of the heat flux vector ($n = 1, 2, 3$).

Undrained Pore Fluid Analysis

POR Field: yes History: yes .fil: no

Pore pressure.

Cohesive Elements

NEEQ
Field: yes History: yes .fil: no

Equivalent nominal strain.

NEEQR
Field: yes History: yes .fil: no

Equivalent nominal strain rate.

MAXSCRT
Field: no History: yes .fil: no

Maximum nominal stress damage initiation criterion.

MAXECRT
Field: no History: yes .fil: no

Maximum nominal strain damage initiation criterion.

QUADSCRT
Field: no History: yes .fil: no

Quadratic nominal stress damage initiation criterion.

QUADECRT
Field: no History: yes .fil: no

Quadratic nominal strain damage initiation criterion.

DMICRT
Field: yes History: yes .fil: no

All active components of the damage initiation criteria.

SDEG
Field: yes History: yes .fil: no

Overall scalar stiffness degradation.

STATUS
Field: yes History: yes .fil: no

Status of the element (the status of an element is 1.0 if the element is active, 0.0 if the element is not).

The status output is added automatically by the analysis.

MMIXDME
Field: yes History: yes .fil: no

Mode mix ratio during damage evolution. It has a value of -1.0 before initiation of damage.

MMIXDMI
Field: yes History: yes .fil: no

Mode mix ratio at damage initiation. It has a value of -1.0 before initiation of damage.

Eulerian Elements

EVF
Field: yes History: yes .fil: no

Eulerian volume fraction. Output includes volume fraction data for each material defined in the Eulerian section, plus the volume fraction of void.

DENSITYVAVG
Field: yes History: no .fil: no

Density, computed as a volume fraction weighted average of all materials in the element.

MISESVAVG Field: yes History: no .fil: no

Mises stress, computed as a volume fraction weighted average of all materials in the element.

PEVAVG Field: yes History: no .fil: no

Plastic strain components, computed as a volume fraction weighted average of all materials in the element.

PEEQVAVG Field: yes History: no .fil: no

Equivalent plastic strain, computed as a volume fraction weighted average of all materials in the element.

PRESSVAVG Field: yes History: no .fil: no

Equivalent pressure stress, computed as a volume fraction weighted average of all materials in the element.

PORVAVG Field: yes History: no .fil: no

Equivalent pore pressure, computed as a volume fraction weighted average of all materials in the element.

SVAVG Field: yes History: no .fil: no

Stress components, computed as a volume fraction weighted average of all materials in the element.

TEMPMAVG Field: yes History: no .fil: no

Temperature, computed as a mass fraction weighted average of all materials in the element.

Element Section Variables

The output variables listed below are available in Abaqus/Explicit.

STH Field: yes History: yes .fil: yes

Section thickness (shell, membrane, and plane stress elements only).

STHIN Field: yes History: yes .fil: no

Section thinning or thickening is defined as $STHIN = 1 - \frac{STH}{STH_{orig}}$, where STH_{orig} is the original thickness specified on the section definition for shell, membrane, and plane stress elements.

SFAILRATIO Field: yes History: no .fil: no

Section failure ratio across all the shell's layers defined as $SFailRatio = \frac{numFailedLayers}{numTotalLayers}$, where $numFailedLayers$ is the number of failed layers (a shell layer is considered failed when all of the section points in the layer failed), and $numTotalLayers$ is the total number of layers in the shell.

SF Field: yes History: yes .fil: yes

All section resultant components, both translational (forces) and rotational (moments).

SF*n* Field: no History: yes .fil: no

Section force per unit width of component *n*, $n = 1, 2, 3, 4, 5$ for conventional shells; $n = 1, 2, 3, 4, 5, 6$ for continuum shells; $n = 1, 2, 3$ for beams and pipes.

SM*n* Field: no History: yes .fil: no

Section moment per unit width of component *n*, $n = 1, 2, 3$.

SORIENT Field: yes History: no .fil: no

Composite shell section orientations.

SE Field: yes History: yes .fil: yes

All section strain, curvature change, and twist components.

SE*n* Field: no History: yes .fil: no

Section nominal strain component *n*, $n = 1, 2, 3, 4, 5, 6$ for shells; $n = 1, 2, 3$ for beams and pipes.

SK*n* Field: no History: yes .fil: no

Section curvature change or twist *n*, $n = 1, 2, 3$.

SSAVG Field: yes History: no .fil: yes

All average membrane and transverse shear stress components (shell elements only).

SSAVG*n* Field: yes History: no .fil: no

Average membrane or transverse shear stress component *n*, $n = 1, 2, 3, 4, 5, 6$ (shell elements only).

References:

- *Element Output*
- *Writing Element Output to the Output Database*

Whole Element Variables

The output variables listed below are available in Abaqus/Explicit.

ELEN Field: yes History: yes .fil: yes

All energy magnitudes in the element.

ELSE Field: yes History: yes .fil: no

Total elastic strain energy in the element (includes energy in transverse shear deformation in shells).

ELCD Field: yes History: yes .fil: no

Total energy dissipated in the element by viscoelastic deformation. (Not supported for hyperelastic and hyperfoam material models with linear viscoelasticity.)

ELPD Field: yes History: yes .fil: no

Total energy dissipated in the element by rate-independent and rate-dependent plastic deformation. For superelastic materials, this variable also includes recoverable phase-transformation energy.

ELVD Field: yes History: yes .fil: no

Total energy dissipated in the element by viscous effects. This includes bulk viscosity and material damping.

ELASE Field: yes History: yes .fil: no

Total "artificial" strain energy in the element. This includes hourglass energy and drilling stiffness energy in shells.

ELIHE Field: yes History: yes .fil: no

Internal heat energy in the element.

ELDMD Field: yes History: yes .fil: no

Total energy dissipated in the element by damage.

ELDC Field: yes History: yes .fil: no

Total energy dissipated in the element by distortion control.

ELEDEN Field: yes History: no .fil: no

All element energy density components.

ESEDEN Field: yes History: no .fil: no

Total elastic strain energy density in the element.

EPDDEN Field: yes History: no .fil: no

Total energy dissipated per unit volume in the element by rate-independent and rate-dependent plastic deformation.

ECDDEN Field: yes History: no .fil: no

Total energy dissipated per unit volume in the element by viscoelasticity.

EVDDEN Field: yes History: no .fil: no

Total energy dissipated per unit volume in the element by viscous effects.

EASEDEN Field: yes History: no .fil: no

Total “artificial” strain energy density in the element (energy associated with constraints used to remove singular modes, such as hourglass control).

EIHEDEN Field: yes History: no .fil: no

Internal heat energy density in the element.

EDMDDEN Field: yes History: no .fil: no

Total energy dissipated per unit volume in the element by damage.

EDCDEN Field: yes History: no .fil: no

Total energy dissipated per unit volume in the element by distortion control.

EDT Field: yes History: yes .fil: yes

Element stable time increment.

EMSF Field: yes History: yes .fil: yes

Element mass scaling factor.

STATUS Field: yes History: yes .fil: yes

Status of the element (material failure with progressive damage, shear failure model, tensile failure model, porous failure criterion, brittle failure model, Johnson-Cook plasticity model, and *VUMAT*). The status of an element is 1.0 if the element is active, 0.0 if the element is not.

The status output is added automatically by the analysis.

EVOL Field: yes History: yes .fil: no

Current element volume. (Only available for continuum and structural elements not using general beam or shell section definitions.)

NFORC Field: yes History: yes .fil: no

Forces at the nodes of an element from both the hourglass and the regular deformation modes of that element (negative of the internal forces in the global coordinate system).

GRAV Field: yes History: no .fil: no

Uniformly distributed gravity load (measured as g, where g is the gravitational acceleration).

SBF Field: yes History: no .fil: no

Stagnation body force.

BF Field: yes History: no .fil: no

Uniformly distributed body force, including viscous body force.

EDMICRTMAX Field: yes History: no .fil: no

Whole shell element maximum damage initiation output among all of the layers, all of the damage initiation criteria, and for fully integrated elements across all of the integration points.

This output variable is the same as DMICRTMAX output for solid and beam elements but complements the DMICRTMAX output variable for composite shell elements because it extracts the maximum damage initiation across all of the layers.

This output variable generates four element output quantities as follows:

EDMICRTMAXVAL outputs the maximum damage initiation value in the entire element.

EDMICRTLAYER outputs the layer number in which the maximum damage initiation value occurred.

EDMICRTTYPE outputs a value that represents the damage initiation criteria type that reached the maximum value in the element, as described in the DMICRTMAX output variable description.

EDMICRTINTP outputs the integration point number for which the maximum damage value occurred. For reduced-integration elements, the output value is one.

The maximum damage initiation output values are retained across the requested output frames until a higher maximum damage initiation value is computed.

References:

- *Element Output*
- *Writing Element Output to the Output Database*

Connector Elements

CTF

Field: yes History: yes .fil: yes

All components of connector total forces and moments.

CTF*n*

Field: no History: yes .fil: no

Connector total force component n ($n = 1, 2, 3$).

CTM*n*

Field: no History: yes .fil: no

Connector total moment component n ($n = 1, 2, 3$).

CEF

Field: yes History: yes .fil: yes

All components of connector elastic forces and moments.

CEF*n*

Field: no History: yes .fil: no

Connector elastic force component n ($n = 1, 2, 3$).

CEM*n*

Field: no History: yes .fil: no

Connector elastic moment component n ($n = 1, 2, 3$).

CUE

Field: yes History: yes .fil: yes

Elastic displacements and rotations in all directions.

CUE*n*

Field: no History: yes .fil: no

Elastic displacement in the n-direction ($n = 1, 2, 3$).

CURE*n*

Field: no History: yes .fil: no

Elastic rotation in the n-direction ($n = 1, 2, 3$).

CUP

Field: yes History: yes .fil: yes

Plastic relative displacements and rotations in all directions.

CUP*n* Field: no History: yes .fil: no

Plastic relative displacement in the *n*-direction ($n = 1, 2, 3$).

CURP*n* Field: no History: yes .fil: no

Plastic relative rotation in the *n*-direction ($n = 1, 2, 3$).

CUPEQ Field: yes History: yes .fil: yes

Equivalent plastic relative displacements and rotations in all directions, and equivalent plastic relative motion for a coupled plasticity definition.

CUPEQ*n* Field: no History: yes .fil: no

Equivalent plastic relative displacement in the *n*-direction ($n = 1, 2, 3$).

CURPEQ*n* Field: no History: yes .fil: no

Equivalent plastic relative rotation in the *n*-direction ($n = 1, 2, 3$).

CUPEQC Field: no History: yes .fil: no

Equivalent plastic relative motion for a coupled plasticity definition.

CALPHAF Field: no History: yes .fil: yes

All components of connector kinematic hardening shift forces and moments.

CALPHAF*n* Field: no History: yes .fil: no

Connector kinematic hardening shift force component *n* ($n = 1, 2, 3$).

CALPHAM*n* Field: no History: yes .fil: no

Connector kinematic hardening shift moment component *n* ($n = 1, 2, 3$).

CVF Field: yes History: yes .fil: yes

All components of connector viscous forces and moments.

CVF*n* Field: no History: yes .fil: no

Connector viscous force component *n* ($n = 1, 2, 3$).

CVM*n* Field: no History: yes .fil: no

Connector viscous moment component *n* ($n = 1, 2, 3$).

CUF Field: yes History: yes .fil: no

All components of connector uniaxial forces and moments.

CUF*n* Field: no History: yes .fil: no

Connector uniaxial force component *n* ($n = 1, 2, 3$).

CUM*n* Field: no History: yes .fil: no

Connector uniaxial moment component *n* ($n = 1, 2, 3$).

CSF Field: no History: yes .fil: yes

All components of connector friction forces and moments.

CSF*n* Field: no History: yes .fil: no

Connector friction force component *n* ($n = 1, 2, 3$).

CSM*n* Field: no History: yes .fil: no

Connector friction moment component *n* ($n = 1, 2, 3$).

CSFC Field: no History: yes .fil: no

Connector friction force in the instantaneous slip direction. Available only if friction is defined in the slip direction.

CNF Field: no History: yes .fil: yes

All components of connector friction-generating contact forces and moments.

CNF*n* Field: no History: yes .fil: no

Connector friction-generating contact force component *n* (*n* = 1, 2, 3).

CNM*n* Field: no History: yes .fil: no

Connector friction-generating contact moment component *n* (*n* = 1, 2, 3).

CNFC Field: no History: yes .fil: no

Connector friction-generating contact force in the instantaneous slip direction. Available only if friction is defined in the slip direction.

CDMG Field: yes History: yes .fil: yes

All components of the overall damage variable.

CDMG*n* Field: no History: yes .fil: no

Overall damage variable component *n* ($n = 1, 2, 3$).

CDMGR*n* Field: no History: yes .fil: no

Overall damage variable component *n* ($n = 1, 2, 3$).

CDIF Field: no History: yes .fil: yes

Components of connector force-based damage initiation criterion in all directions.

CDIF*n* Field: no History: yes .fil: no

Connector force-based damage initiation criterion in the *n*-translation direction ($n = 1, 2, 3$).

CDIFR*n* Field: no History: yes .fil: no

Connector force-based damage initiation criterion in the *n*-rotation direction ($n = 1, 2, 3$).

CDIFC Field: no History: yes .fil: no

Connector force-based damage initiation criterion in the instantaneous slip direction.

CDIM Field: no History: yes .fil: yes

Components of connector motion-based damage initiation criterion in all directions.

CDIM*n* Field: no History: yes .fil: no

Connector motion-based damage initiation criterion in the *n*-translation direction ($n = 1, 2, 3$).

CDIMR*n* Field: no History: yes .fil: no

Connector motion-based damage initiation criterion in the *n*-rotation direction ($n = 1, 2, 3$).

CDIMC Field: no History: yes .fil: no

Connector motion-based damage initiation criterion in the instantaneous slip direction.

CDIP Field: yes History: yes .fil: yes

Components of connector plastic motion-based damage initiation criterion in all directions (including the instantaneous slip direction).

CDIP*n* Field: no History: yes .fil: no

Connector plastic motion-based damage initiation criterion in the *n*-translation direction ($n = 1, 2, 3$).

CDIPR*n* Field: no History: yes .fil: no

Connector plastic motion-based damage initiation criterion in the *n*-rotation direction ($n = 1, 2, 3$).

CDIPC Field: no History: yes .fil: no

Connector plastic motion-based damage initiation criterion in the instantaneous slip direction.

CSLST Field: no History: yes .fil: yes

All flags for connector stop and connector lock status.

CSLST*i* Field: no History: yes .fil: no

Flag for connector stop and connector lock status in the *i*-direction ($i = 1, \ldots, 6$).

CASU Field: no History: yes .fil: yes

Components of accumulated slip in all directions.

CASU*n* Field: no History: yes .fil: no

Connector accumulated slip in the *n*-direction (*n* = 1, 2, 3).

CASUR*n* Field: no History: yes .fil: no

Connector angular accumulated slip in the *n*-direction (*n* = 1, 2, 3).

CASUC Field: no History: yes .fil: no

Connector accumulated slip in the instantaneous slip direction. Available only if friction is defined in the slip direction.

CIVC Field: no History: yes .fil: yes

Connector instantaneous velocity in the slip direction. Available only if friction is defined in the slip direction.

CRF Field: no History: yes .fil: yes

All components of connector reaction forces and moments.

CRF*n* Field: no History: yes .fil: no

Connector reaction force component *n* ($n = 1, 2, 3$).

CRM*n* Field: no History: yes .fil: no

Connector reaction moment component *n* ($n = 1, 2, 3$).

CCF Field: no History: yes .fil: yes

All components of connector concentrated forces and moments.

CCF*n* Field: no History: yes .fil: no

Connector concentrated force component *n* ($n = 1, 2, 3$).

CCM*n* Field: no History: yes .fil: no

Connector concentrated moment component *n* ($n = 1, 2, 3$).

CP Field: yes History: yes .fil: yes

Relative positions in all directions.

CP*n* Field: no History: yes .fil: no

Relative position in the *n*-direction ($n = 1, 2, 3$).

CPR*n* Field: no History: yes .fil: no

Relative angular position in the *n*-direction ($n = 1, 2, 3$).

CU Field: yes History: yes .fil: yes

Relative displacements and rotations in all directions.

CU*n* Field: no History: yes .fil: no

Relative displacement in the *n*-direction ($n = 1, 2, 3$).

CUR*n* Field: no History: yes .fil: no

Relative rotation in the *n*-direction ($n = 1, 2, 3$).

CCU Field: no History: yes .fil: yes

Constitutive displacements and rotations in all directions.

CCU*n* Field: no History: yes .fil: no

Constitutive displacement in the *n*-direction ($n = 1, 2, 3$).

CCUR*n* Field: no History: yes .fil: no

Constitutive rotation in the *n*-direction ($n = 1, 2, 3$).

CV Field: yes History: yes .fil: yes

Relative velocities in all directions.

CV*n* Field: no History: yes .fil: no

Relative velocity in the *n*-direction ($n = 1, 2, 3$).

CVR*n* Field: no History: yes .fil: no

Relative angular velocity in the *n*-direction ($n = 1, 2, 3$).

CA Field: yes History: yes .fil: yes

Relative accelerations in all directions.

CA*n* Field: no History: yes .fil: no

Relative acceleration in the *n*-direction ($n = 1, 2, 3$).

CAR*n* Field: no History: yes .fil: no

Relative angular acceleration in the *n*-direction ($n = 1, 2, 3$).

CFAILST Field: yes History: yes .fil: yes

All flags for connector failure status.

CFAILST*i* Field: no History: yes .fil: no

Flag for connector failure status in the *i*-direction ($i = 1, \ldots, 6$).

CDERU Field: yes History: yes .fil: no

Connector derived displacement.

CDERF Field: yes History: yes .fil: no

Connector derived force.

Particle Elements

SMOOTHLEN Field: yes History: no .fil: no

Smoothing length of continuum particle elements with SPH formulation.

Element Face Variables

The output variables listed below are available in Abaqus/Explicit.

P Field: yes History: no .fil: no

Uniformly distributed pressure load on element faces. When the pressure is defined using **DLOAD*, the variable name is changed automatically to PDLOAD.

STAGP Field: yes History: no .fil: no

Stagnation pressure load on element faces.

VP Field: yes History: no .fil: no

Viscous pressure load on element faces.

IWCONWEP Field: yes History: no .fil: no

Air blast pressure load from the CONWEP model on element faces.

TRNOR Field: yes History: no .fil: no

Normal component (component along face normal) of traction load on element faces.

TRSHR Field: yes History: no .fil: no

Shear component (component along face tangent) of traction load on element faces.

References:

- *Writing Element Output to the Output Database*

Nodal Variables

The output variables listed below are available in Abaqus/Explicit.

COORD Field: yes History: yes .fil: yes

Coordinates of the node. These are the current coordinates if the large-displacement formulation is being used.

COOR*n* Field: no History: yes .fil: no

Coordinate *n* ($n = 1, 2, 3$).

U Field: yes History: yes .fil: yes

Displacement components.

Results file and field-type output: both translation and rotation.

History-type output: translation only. Rotation results should be requested by components.

UT Field: yes History: yes .fil: no

Translational displacement components.

UMAG Field: no History: yes .fil: no

Magnitude of translational displacements.

UR Field: yes History: yes .fil: no

Rotational displacement components.

U*n* Field: no History: yes .fil: no

u_n displacement component ($n = 1, 2, 3$).

UR*n* Field: no History: yes .fil: no

ϕ_n rotation component ($n = 1, 2, 3$).

V Field: yes History: yes .fil: yes

Velocity components (both translation and rotation).

Results file and field-type output: both translation and rotation.

History-type output: translation only. Rotation results should be requested by components.

VT Field: yes History: yes .fil: no

Translational velocity components.

VMAG Field: no History: yes .fil: no

Magnitude of translational velocities.

VR Field: yes History: yes .fil: no

Rotational velocity components.

V*n* Field: no History: yes .fil: no

$\dot{u}_n$ velocity component ($n = 1, 2, 3$).

VR*n* Field: no History: yes .fil: no

$\dot{\phi}_n$ rotational velocity component ($n = 1, 2, 3$).

A Field: yes History: yes .fil: yes

Acceleration components (both translation and rotation).

Results file and field-type output: both translation and rotation.

History-type output: translation only. Rotation results should be requested by components.

AT Field: yes History: yes .fil: no

Translational acceleration components.

AMAG Field: no History: yes .fil: no

Magnitude of translational accelerations.

AR Field: yes History: yes .fil: no

Rotational acceleration components.

A*n* Field: no History: yes .fil: no

$\ddot{u}_n$ acceleration component ($n = 1, 2, 3$).

AR*n* Field: no History: yes .fil: no

$\ddot{\phi}_1$ rotational acceleration component ($n = 1, 2, 3$).

POR Field: yes History: yes .fil: yes

Acoustic pressure at a node.

PABS Field: yes History: yes .fil: yes

Acoustic absolute pressure at a node.

NT Field: yes History: yes .fil: yes

All temperature values at a node. Available only for coupled thermal-stress analysis.

NT*n* Field: no History: yes .fil: no

Temperature degree of freedom *n* at a node ($n = 11$). Available only for coupled thermal-stress analysis.

RF Field: yes History: yes .fil: yes

Reaction force and moment components.

Results file and field-type output: both translation and rotation.

History-type output: translation only. Rotation results should be requested by components.

RT Field: yes History: yes .fil: no

Reaction force components.

RFMAG Field: no History: yes .fil: no

Magnitude of reaction forces.

RM Field: yes History: yes .fil: no

Reaction moment components.

RF*n* Field: no History: yes .fil: no

Reaction force component *n* ($n = 1, 2, 3$) (conjugate to prescribed displacement u_n).

RFL Field: yes History: yes .fil: yes

All reaction flux values. Available only for coupled thermal-stress analysis.

RFL*n* Field: yes History: yes .fil: no

Reaction flux value *n* at a node ($n = 11$). Available only for coupled thermal-stress analysis.

RM*n* Field: no History: yes .fil: no

Reaction moment component *n* ($n = 1, 2, 3$) (conjugate to prescribed rotation ϕ_n).

CF Field: yes History: yes .fil: no

All components of point loads and concentrated moments.

CF*n* Field: no History: yes .fil: no

Point load component *n* ($n = 1, 2, 3$).

CM*n* Field: no History: yes .fil: no

Point moment component *n* ($n = 1, 2, 3$).

FEXT Field: yes History: yes .fil: no

All components of external point loads from a co-simulation or external field definition.

MEXT Field: yes History: yes .fil: no

All components of external point moments from a co-simulation or external field definition.

NVF Field: yes History: no .fil: no

Nodal volume fraction.

STRAINFREE Field: yes History: no .fil: no

Strain-free adjustments to initial positions (adjusted position minus unadjusted position). Only written to the output database (`.odb`) file for the original field output frame at zero time.

TIEDSTATUS Field: yes History: no .fil: no

Status of the tied secondary nodes (the status of a secondary node is 2 if the secondary node is not tied, 1 if the secondary node is tied, and 0 for nodes that do not participate in a tie constraint).

TIEADJUST Field: yes History: no .fil: no

Position adjustment vector components of the tied secondary nodes. Only written to the output database (`.odb`) file for the original field output frame at zero time.

PRELV Field: yes History: no .fil: no

Discrete particle relative velocity with respect to the fluid (available for Abaqus/Standard to Abaqus/Explicit co-simulation.)

References:

- *Node Output*
- *Writing Nodal Output to the Output Database*

Fluid Cavity Variables

PCAV

Field: no History: yes .fil: yes

Fluid cavity gauge pressure.

CVOL

Field: no History: yes .fil: yes

Fluid cavity volume.

CTEMP

Field: no History: yes .fil: no

Fluid cavity temperature for an ideal gas model used under adiabatic conditions.

CSAREA

Field: no History: yes .fil: no

Fluid cavity surface area.

CLAREA

Field: no History: yes .fil: no

Fluid cavity unblocked leakage area.

CBLARAT

Field: no History: yes .fil: no

Ratio of the blocked leakage area to the unblocked leakage area.

CMASS

Field: no History: yes .fil: no

Mass of the fluid contained in a fluid cavity.

APCAV

Field: no History: yes .fil: no

Average gauge pressures for multiple fluid cavities.

TCVOL

Field: no History: yes .fil: no

Total volume of multiple fluid cavities.

ACTEMP

Field: no History: yes .fil: no

Average fluid cavity temperature for an ideal gas model used under adiabatic conditions for multiple fluid cavities.

TCSAREA

Field: no History: yes .fil: no

Total surface area of multiple fluid cavities.

TCMASS

Field: no History: yes .fil: no

Total mass of the fluid contained in the multiple fluid cavities.

CMF

Field: no History: yes .fil: no

Molecular mass fraction of fluid species contained in a fluid cavity.

CMFL

Field: no History: yes .fil: no

Mass flow rate out of a fluid cavity.

CMFLT Field: no History: yes .fil: no

Accumulated mass flow out of a fluid cavity.

CEFL Field: no History: yes .fil: no

Heat energy flow rate out of a fluid cavity.

CEFLT Field: no History: yes .fil: no

Accumulated heat energy flow out of a fluid cavity.

MINFL Field: no History: yes .fil: no

Inflator mass flow rate into a fluid cavity.

MINFLT Field: no History: yes .fil: no

Accumulated inflator mass flow into a fluid cavity.

TINFL Field: no History: yes .fil: no

Inflator temperature.

Surface Variables

The output variables listed below are available in Abaqus/Explicit.

References:

- *Writing Surface Output to the Output Database*
- *About General Contact in Abaqus/Explicit*
- *About Contact Pairs in Abaqus/Explicit*
- *Thermal Contact Properties*

Mechanical Analysis–Nodal Quantities

CFORCE Field: yes History: no .fil: no

Contact normal force (CNORMF) and frictional shear force (CSHEARF).

CDISP Field: yes History: no .fil: no

Contact opening (COPEN) and accumulated tangential motions (CSLIP1, CSLIP2, and CSLIPEQ) for general contact analyses.

CEDGEACTIVE Field: yes History: no .fil: no

Status of contact edges for general contact analyses (active as primary, active as secondary and deactive).

CFRICWORK Field: yes History: no .fil: no

Contact frictional work for general contact analyses.

CNAREA Field: yes History: no .fil: no

Contact nodal area for each node with active contact forces in general contact, based on area contributions from adjacent surface facets projected to a plane perpendicular to the current nodal contact normal force vector.

CNMSF Field: yes History: no .fil: no

Nodal contact mass scaling factor.

COPENMIN Field: yes History: no .fil: no

Minimum contact normal distance for general contact analyses.

CORIENT Field: yes History: no .fil: no

Surface tangent directions (CORIENT1 and CORIENT2) for general contact analyses.

CSLIPR Field: yes History: no .fil: no

Instantaneous contact slip rates (CSLIPR1, CSLIPR2, and CSLIPRMAG) for general contact analyses.

CSTATUS Field: yes History: no .fil: no

Contact status for general contact analyses.

CSTRESS Field: yes History: no .fil: no

Contact pressure (CPRESS) and magnitude of the frictional shear stress (CSHEARMAG) for general contact analyses.

Contact pressure (CPRESS) and frictional shear stresses (CSHEAR1 and CSHEAR2) for contact pair analyses.

CTANDIR

Field: yes History: no .fil: no

Instantaneous contact tangent directions (CTANDIR1 and CTANDIR2) for general contact analyses.

CTHICK

Field: yes History: no .fil: no

Contact thickness for general contact analyses.

CSMAXSCRT

Field: yes History: no .fil: no

Maximum value of the maximum stress-based damage initiation criterion for cohesive surfaces in general contact up to the current increment.

CSQUADSCRT

Field: yes History: no .fil: no

Maximum value of the quadratic stress-based damage initiation criterion for cohesive surfaces in general contact up to the current increment.

CSMAXUCRT

Field: yes History: no .fil: no

Maximum value of the maximum separation-based damage initiation criterion for cohesive surfaces in general contact up to the current increment.

CSQUADUCRT

Field: yes History: no .fil: no

Maximum value of the quadratic separation-based damage initiation criterion for cohesive surfaces in general contact up to the current increment.

CSDMG

Field: yes History: no .fil: no

Damage variable for cohesive surfaces in general contact.

CWEAR

Field: yes History: no .fil: no

Accumulated nodal wear vector for general contact analyses.

FSLIP

Field: yes History: no .fil: no

Length of contact slip path at secondary nodes during contact (FSLIPEQ) and in some cases (see *About Contact Pairs in Abaqus/Explicit*) components of net contact slip in local tangent directions (FSLIP1 and FSLIP2). These variables remain constant while a secondary node is not in contact.

FSLIPR

Field: yes History: no .fil: no

Magnitude of contact slip rate at secondary nodes during contact (FSLIPR) and in some cases (see *About Contact Pairs in Abaqus/Explicit*) components of contact slip rate in local tangent directions (FSLIPR1 and FSLIPR2). These variables are set to zero while a secondary node is not in contact.

BONDSTAT

Field: no History: yes .fil: no

Spot weld bond status.

BONDLOAD

Field: no History: yes .fil: no

Spot weld bond load.

CFORCEC

Field: yes History: no .fil: no

Consistent contact normal force (CNORMFC) and frictional shear force (CSHEARFC) in general contact with active C3D10 elements.

NDTTOTAL
Field: yes History: no .fil: no

Stable time increment over contact surfaces with effects of element and contact mass scaling.

PPRESS
Field: yes History: no .fil: no

Pressure due to fluid pressure penetration loading.

PFORCE
Field: yes History: no .fil: no

Force due to fluid pressure penetration loading.

Crack Bond Failure Quantities

DBT
Field: yes History: no .fil: no

Time when bond failure occurs.

DBS
Field: yes History: no .fil: no

All components of remaining stress in the failed bond.

DBSF
Field: yes History: no .fil: no

Fraction of stress that remains at bond failure.

BDSTAT
Field: yes History: no .fil: no

Bond state (the state is 1.0 if bonded, 0.0 if unbonded).

OPENBC
Field: yes History: no .fil: no

Relative displacement behind crack when fracture criterion is met.

CRSTS
Field: yes History: no .fil: no

All components of critical stress at failure.

ENRRT
Field: yes History: no .fil: no

All components of strain energy release rate.

EFENRRTR
Field: yes History: no .fil: no

Effective energy release rate ratio.

Mechanical Analysis–Whole Surface Quantities

CFN
Field: no History: yes .fil: no

Total force due to contact pressure (CFNn, n = 1, 2, 3).

CFNM
Field: no History: yes .fil: no

Magnitude of total force due to contact pressure.

CFS
Field: no History: yes .fil: no

Total force due to frictional stress (CFSn, n = 1, 2, 3).

CFSM
Field: no History: yes .fil: no

Magnitude of total force due to frictional stress.

CFT Field: no History: yes .fil: no

Total force due to contact pressure and frictional stress (CFTn, n = 1, 2, 3).

CFTM Field: no History: yes .fil: no

Magnitude of total force due to contact pressure and frictional stress.

CMN Field: no History: yes .fil: no

Total moment about the origin due to contact pressure (CMNn, n = 1, 2, 3).

CMNM Field: no History: yes .fil: no

Magnitude of total moment about the origin due to contact pressure.

CMS Field: no History: yes .fil: no

Total moment about the origin due to frictional stress (CMSn, n = 1, 2, 3).

CMSM Field: no History: yes .fil: no

Magnitude of total moment about the origin due to frictional stress.

CMT Field: no History: yes .fil: no

Total moment about the origin due to contact pressure and frictional stress (CMTn, n = 1, 2, 3).

CMTM Field: no History: yes .fil: no

Magnitude of total moment about the origin due to contact pressure and frictional stress.

CAREA Field: no History: yes .fil: no

Total area in contact.

XN Field: no History: yes .fil: no

Center of the total force due to contact pressure (XNn, n = 1, 2, 3).

XS Field: no History: yes .fil: no

Center of the total force due to frictional stress (XSn, n = 1, 2, 3).

XT Field: no History: yes .fil: no

Center of the total force due to contact pressure and frictional stress (XTn, n = 1, 2, 3).

PFN Field: no History: yes .fil: no

Total force due to fluid pressure penetration loading (PFNn, n = 1, 2, 3).

PFNM Field: no History: yes .fil: no

Magnitude of total force due to fluid pressure penetration loading.

Fully Coupled Temperature-Displacement Analysis

HFL Field: yes History: no .fil: no

Heat flux per unit area leaving the surface.

HFLA Field: yes History: no .fil: no

HFL multiplied by the nodal area.

HTL Field: yes History: no .fil: no

Time integrated HFL.

HTLA Field: yes History: no .fil: no

HTL multiplied by the nodal area.

SFDR Field: yes History: no .fil: no

Heat flux per unit area due to frictional dissipation.

SFDRA Field: yes History: no .fil: no

SFDR multiplied by the nodal area.

SFDRT Field: yes History: no .fil: no

Time integrated SFDR.

SFDRTA Field: yes History: no .fil: no

SFDRT multiplied by the nodal area.

Integrated Variables

The output variables listed below are available in Abaqus/Explicit.

SOAREA Field: no History: yes .fil: no

Area of the surface as projected onto a plane normal to the average surface normal.

SOF Field: no History: yes .fil: no

Total force transmitted through the surface.

SOM Field: no History: yes .fil: no

Total moment transmitted through the surface. The moment of the forces transmitted through the surface is taken about the current location of the reference node if one is specified on an integrated output section and is associated with the integrated output request. The moment is taken about the global origin either if no section definition is associated with the integrated output request or if there is no reference node defined in the associated section definition.

MASS Field: no History: yes .fil: no

Total mass of the element set.

DMASS Field: no History: yes .fil: no

Total mass change in percentage of the element set due to mass scaling.

UCOM Field: no History: yes .fil: no

Equivalent rigid-body translational displacement of the element set.

VCOM Field: no History: yes .fil: no

Equivalent rigid-body translational velocity of the element set.

ACOM Field: no History: yes .fil: no

Equivalent rigid-body translational acceleration of the element set.

COORDCOM Field: no History: yes .fil: no

Coordinates of the center of mass of the element set.

MASSEUL Field: no History: yes .fil: no

Total mass of each Eulerian material instance in the element set.

VOLEUL Field: no History: yes .fil: no

Total volume of each Eulerian material instance in the element set.

PAVG Field: no History: yes .fil: no

Pressure averaged over the total volume of each Eulerian material instance in the element set.

TAVG Field: no History: yes .fil: no

Temperature averaged over the total mass of each Eulerian material instance in the element set.

PDMASS Field: no History: yes .fil: no

Generated mass of a discrete particle element set.

PDVRMS Field: no History: yes .fil: no

RMS velocity of a discrete particle element set.

PDTEMP Field: no History: yes .fil: no

Average temperature of a lumped gas particle species.

PDPAVG Field: no History: yes .fil: no

Average pressure of a lumped gas particle species.

PDPGAUGE Field: no History: yes .fil: no

Average gauge pressure of a lumped gas particle species.

References:

- *Integrated Output*
- *Integrated Output Section Definition*

Total Energy Output

The output variables listed below are available in Abaqus/Explicit.

ALLAE

Field: no History: yes .fil: yes

“Artificial” strain energy associated with constraints used to remove singular modes (such as hourglass control) and with constraints used to make the drill rotation follow the in-plane rotation of the shell elements.

ALLCD

Field: no History: yes .fil: yes

Energy dissipated by viscoelasticity. (Not supported for hyperelastic and hyperfoam material models with linear viscoelasticity.)

ALLFD

Field: no History: yes .fil: yes

Total energy dissipated through frictional effects. (Available only for the whole model).

ALLIE

Field: no History: yes .fil: yes

Total strain energy. (ALLIE=ALLSE + ALLPD + ALLCD + ALLAE + ALLDMD+ ALLDC+ ALLFC.)

ALLKE

Field: no History: yes .fil: yes

Kinetic energy.

ALLPD

Field: no History: yes .fil: yes

Energy dissipated by rate-independent and rate-dependent plastic deformation. For superelastic materials, this variable also includes recoverable phase-transformation energy.

ALLSE

Field: no History: yes .fil: yes

Recoverable strain energy.

ALLVD

Field: no History: yes .fil: yes

Energy dissipated by viscous effects not including externally applied viscous pressure loads or viscous body force loads.

ALLWK

Field: no History: yes .fil: yes

External work by applied loads including externally applied viscous pressure loads and viscous body force loads. (Available only for the whole model).

ALLIHE

Field: no History: yes .fil: yes

Internal heat energy.

ALLHF

Field: no History: yes .fil: yes

External heat energy through external fluxes.

ALLDMD

Field: no History: yes .fil: yes

Energy dissipated by damage.

ALLDC

Field: no History: yes .fil: yes

Energy dissipated by distortion control.

ALLFC Field: no History: yes .fil: no

Fluid cavity energy, defined as the negative of the work done by all fluid cavities. (Available only for the whole model.)

ALLPW Field: no History: yes .fil: no

Work done by contact penalties, including general contact and penalty/kinematic contact pairs. (Available only for the whole model.)

ALLCW Field: no History: yes .fil: no

Work done by constraint penalties. (Available only for the whole model.)

ALLMW Field: no History: yes .fil: no

Work done in propelling mass added in mass scaling. (Available only for the whole model.)

ALLPG Field: no History: yes .fil: no

Energy added by particle generators. (Available only for the whole model.)

ETOTAL Field: no History: yes .fil: yes

Energy balance defined as: ALLKE + ALLIE + ALLVD + ALLFD + ALLIHE – ALLWK – ALLPG – ALLPW – ALLCW – ALLMW – ALLHF. (Available only for the whole model.)

References:

- *Total Energy Output*
- *Total Energy Output*

Time Increment and Mass Output

The output variables listed below are available in Abaqus/Explicit.

CDMASS Field: no History: yes .fil: no

Percent change in mass of the model due to contact mass scaling.

DT Field: no History: yes .fil: yes

Time increment.

DMASS Field: no History: yes .fil: yes

Percent change in mass of the model due to contact and element mass scaling.

SSPEEQ Field: no History: yes .fil: no

Steady-state equivalent plastic strain norms.

SSPEEQ*n* Field: no History: yes .fil: no

Steady-state equivalent plastic strain norm *n*.

SSSPRD Field: no History: yes .fil: no

Steady-state spread strain norms.

SSSPRD*n* Field: no History: yes .fil: no

Steady-state spread norm *n*.

SSFORC Field: no History: yes .fil: no

Steady-state force norms.

SSFORC*n* Field: no History: yes .fil: no

Steady-state force norm *n*.

SSTORQ Field: no History: yes .fil: no

Steady-state torque norms.

SSTORQ*n* Field: no History: yes .fil: no

Steady-state torque norm *n*.

References:

- *Output to the Abaqus/Explicit Results File*
- *Time Incrementation Output in Abaqus/Explicit*

The Postprocessing Calculator

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Output to the Output Database*
- *Abaqus/Standard Output Variable Identifiers*
- *Abaqus/Explicit Output Variable Identifiers*
- *Abaqus/Standard and Abaqus/Explicit Execution*

Overview

The postprocessing calculator can perform operations on output quantities written to the output database (*job-name*.`odb`) by Abaqus. It then expands the output database by writing these new output quantities to the output database. Once this expansion is done, it is not possible to convert the output database back to its original form. The postprocessing calculator is for use only with the Visualization module of Abaqus/CAE (Abaqus/Viewer).

Functionality of the Calculator

The postprocessing calculator performs the following calculations on data written to the output database:

- Extrapolation of integration point quantities to the nodes or interpolation of integration point quantities to the centroid of an element, according to the user-specified position for element output; see *Writing Element Output to the Output Database* for details.
- Calculation of history output at tracer particles; see *Tracer Particle Output from Abaqus/Explicit*.

Running the Calculator

The postprocessing calculator is generally not required in Abaqus/Standard because by default Abaqus/Standard performs results postprocessing during the course of the analysis. You can override this default behavior by using the environment variable **auto_calculate** in the Abaqus environment file. See *Environment File Settings* for details.

By default in Abaqus/Explicit or if requested in Abaqus/Standard (using **auto_calculate** in the Abaqus environment file), the postprocessing calculator will run automatically upon the completion of an analysis. During the execution of the analysis, Abaqus will determine if there are keywords in the input file that require the use of the calculator and will initiate the calculator upon completion if it is required. You can override this default behavior by using the environment variable **auto_calculate** in the Abaqus environment file. See *Environment File Settings* for details.

You can run the postprocessing calculator manually by using the **convert**=`odb` option on the **abaqus** execution procedure.

To see the postprocessed results before an analysis is complete, you can run the postprocessing calculator manually while the analysis is still running, using the **oldjob** option in conjunction with the **convert**=`odb` option on the **abaqus** execution procedure. The postprocessing calculator will write a new output database using the value of the **job** parameter as the file name. Due to the fact that the analysis is writing to the output database at the same time the postprocessing calculator is attempting to read it, the output database may be in an inconsistent state that makes reading it impossible. If this problem occurs, the postprocessing calculator will stop attempting to read the output database and exit. A warning message explaining what has happened will be output to the screen. You can then attempt to run the postprocessing calculator again. If the inconsistent state has cleared, the postprocessing calculator will run normally.

If the postprocessing calculator is run during an analysis without the **oldjob** option, Abaqus will ask you to confirm that the existing output database can be overwritten. You should make sure the analysis is complete before running the postprocessing calculator manually without the **oldjob** option. If the analysis is still running when the postprocessing calculator is run without using the **oldjob** option, the output database will be corrupted.

For a detailed description of the procedure for running the postprocessing calculator manually, see *Abaqus/Standard and Abaqus/Explicit Execution*.

If an analysis exits because available CPU time has expired and you restart the analysis, the postprocessing calculator will not automatically expand the output database from the original,stopped run. You must manually run the postprocessing calculator to expand the original output database using the procedure outlined above.

File Output Format

In this section:

About the Results File

Overview

The Abaqus results file is the medium through which analysis results can be carried over into other software, such as postprocessing programs.

Writing Information to the Results File

The following types of output can be written to the results file:

- element output, nodal output, energy output, modal output, contact surface output, and section output (see *Output to the Data and Results Files*)
- element matrix output (see *Element Matrix Output in Abaqus/Standard*)
- substructure matrix output (see *Writing the Recovery Matrix, Reduced Stiffness Matrix, Mass Matrix, Load Case Vectors, and Gravity Vectors to a File*)
- cavity radiation view factor matrices (see *Writing the View Factor Matrices to the Results File*)

About Output describes the general format of the results file.

An Abaqus model can be defined in terms of an assembly of part instances (see *Assembly Definition*). However, the results file is not organized by part; it contains internal node and element numbers (see *About Output*). A map between the original numbers and part instance names and the internal numbers is written to the data file.

Accessing Information in the Results File

This chapter contains technical descriptions of the results file and is intended to be read by users or programmers who need to write programs that use the results file.

- *Results File* describes the format of the individual records in the results file.
- *Accessing the Results File Information* describes the subroutine calls required to read the file output, contains an example of a program written to use the Abaqus results file, and shows how you can write (or modify) a results file using the Abaqus file format.
- *Utility Routines for Accessing the Results File* describes the utility subroutines that can be used to access the results file.

Results File Output Format

In this section:

Results File

Products: Abaqus/Standard Abaqus/Explicit

Overview

This section describes the format of the individual records in the Abaqus results file.

Where applicable, the output variable identifier used in writing a given value to the file is printed below the corresponding record type description. Records that are available only in Abaqus/Standard are designated with an $^{(S)}$; records that are available only in Abaqus/Explicit are designated with an $^{(E)}$. The record key for a particular record may differ between Abaqus/Standard and Abaqus/Explicit.

References:

- *About the Results File*
- *Abaqus/Standard Output Variable Identifiers*
- *Abaqus/Explicit Output Variable Identifiers*

Record Format

The results file is written as a sequential file. Each record has the following format:

Location	Length	Description
1	1	Record length (NW)
2	1	Record type key
3, 4...	($NW - 2$)	Attributes

All words in the results file are of the same length, whether they contain integer, floating point number, or character string data. The word length is that of a double precision floating point number (8 bytes).

The attributes in a given record may depend on the element type being considered. For example, the stress components associated with three-dimensional shell elements are σ_{11}, σ_{22}, and σ_{12} (in local directions), while those associated with three-dimensional solids are σ_{xx}, σ_{yy}, σ_{zz}, σ_{xy}, σ_{xz}, and σ_{yz} (in global directions if no local orientation is specified). Thus, care must be used in interpreting the data when postprocessing the file output. Refer to *Abaqus Elements Guide* for a definition of the ordering of element-dependent attributes.

In steady-state dynamic analyses, complex values are stored as the real components followed by the imaginary components. For example, the stress components associated with three-dimensional shell elements are $\Re(\sigma_{11})$, $\Re(\sigma_{22})$, and $\Re(\sigma_{12})$ followed by $\Im(\sigma_{11})$, $\Im(\sigma_{22})$, and $\Im(\sigma_{12})$.

In models that are defined in terms of an assembly of part instances, the results file contains internal (global) node and element numbers, as explained in *About Output*. Part and assembly records are not included in the results file.

Local Coordinate System

If the components of an element quantity are in local directions, a record of type 85 defining these directions is generated for each point at which component output is requested if the local coordinate directions were requested in Abaqus/Standard (see *Output of Local Directions to the Results File*) and automatically in Abaqus/Explicit. The local coordinate system may be inherent to the element, as is the case in shells and membranes, or may have been defined by a local orientation (see *Orientations*).

For shell elements a direction record is written for every material point in the section for which component output is requested, and a separate direction record is written for section forces and section strains. For geometrically nonlinear analysis in Abaqus/Standard the record contains the current, updated directions, except for small-strain shells, in which case the original directions are given. Direction output is not provided for trusses, two-dimensional beams, axisymmetric shells or membranes, or for values averaged at nodes.

Label Record

Some record types include labels, such as element and node set names, written in A8 format. If a label exceeds 8 characters, an integer identifier will be written instead. This identifier can then be used to cross-reference the actual label stored in 10A8 format on record type 1940.

Records Written for Any File Output Request

This section describes the format of the individual records in the Abaqus results file.

Record Format: Records Written for Any File Output Request

Record key	Record type	Attributes
1900	Element definitions	**1.** Element number. **2.** Element type (characters, A8 format, left justified). **3.** First node on the element. **4.** Second node on the element. **5.** Etc.
1990(S)	Element definition continuation	**1.** Node on the element in the previous 1900 record. **2.** Etc.
1901	Node definitions	**1.** Node number. **2.** First coordinate. **3.** Second coordinate. **4.** Etc.
1902	Active degrees of freedom	**1.** Location in nodal arrays of degree of freedom 1 (0 if DOF 1 is not active in the model). **2.** Location in nodal arrays of degree of freedom 2 (0 if DOF 2 is not active in the model). **3.** Etc.
1910(S)	Substructure path	**1.** 0 substructure enter record; 1 substructure leave record. **2.** Element number on usage level. **3.** Substructure type identifier (Z*n*). **4.** Element number at the previous level if it is not the usage level. **5.** Etc.
1911	Output request definition	**1.** Flag for element-based output (0), nodal output (1), modal output (2), or element set energy output (3). **2.** Set name (node or element set) used in the request (A8 format). This attribute is blank if no set was specified.

		3. Element type (only for element output, A8 format).
1921	Abaqus release, etc.	**1.** Abaqus release number (A8 format). **2.** Date (2A8 format). **3.** Date cont'd. **4.** Time (A8 format). **5.** Number of elements in the model. **6.** Number of nodes in the model. **7.** Typical element length in the model.
1922	Heading	**1.** Attributes 1–10. The heading entered as the first data line of the **HEADING* option (A8 format). Equivalent to the job description in Abaqus/CAE.
1931	Node set	**1.** Node set name (A8 format). In Abaqus/Explicit only node sets defined as part of the model definition are written. **2.** First node in the node set. **3.** Second node in the node set. **4.** Etc.
1932	Node set continuation	**1.** Node number in the node set of the previous 1931 record. **2.** Etc.
1933	Element set	**1.** Element set name (A8 format). In Abaqus/Explicit only element sets defined as part of the model definition are written. **2.** First element in the element set. **3.** Second element in the element set. **4.** Etc.
1934	Element set continuation	**1.** Element number in the element set of the previous 1933 record. **2.** Etc.
1940	Label cross-reference	**1.** Integer reference. **2.** Label (10A8 format).

Record Written Once per Eigenvalue in Natural Frequency Extraction

This section describes the format of the individual records in the Abaqus results file.

Record Format: Record Written Once per Eigenvalue in Natural Frequency Extraction

Record key	Record type	Attributes
1980(S)	Modal	**1.** Eigenvalue number. **2.** Eigenvalue. **3.** Generalized mass. **4.** Composite damping. **5.** Participation factor for degree of freedom 1. **6.** Effective mass for degree of freedom 1. **7.** Participation factor for degree of freedom 2. **8.** Effective mass for degree of freedom 2. **9.** Etc.

Records Written Once per Increment

This section describes the format of the individual records in the Abaqus results file.

Record Format: Records Written Once per Increment

Record key	Record type	Attributes
2000	Increment start record	**1.** Total time. **2.** Step time. **3.** Maximum creep strain-rate ratio (control of solution-dependent amplitude) in Abaqus/Standard; currently not used in Abaqus/Explicit. **4.** Solution-dependent amplitude in Abaqus/Standard; currently not used in Abaqus/Explicit. **5.** Procedure type: gives a key to the step type. See *Table 1* at the end of this section. **6.** Step number. **7.** Increment number. **8.** Linear perturbation flag in Abaqus/Standard: 0 if general step, 1 if linear perturbation step; currently not used in Abaqus/Explicit. **9.** Load proportionality factor: nonzero only in static Riks steps; currently not used in Abaqus/Explicit. **10.** Frequency (cycles/time) in a steady-state dynamic response analysis or steady-state transport angular velocity (rad/time) in a steady-state transport analysis; currently not used in Abaqus/Explicit. **11.** Time increment. **12.** Attributes 12–21. The step subheading entered as the first data line of the **STEP* option (A8 format). Equivalent to the step description in Abaqus/CAE.
2001	Increment end record	**1.** No attributes.

Records Written for Any Element File Output Request

This section describes the format of the individual records in the Abaqus results file.

These records contain data about element variables at integration points within the elements, at the centroid of elements, or at the nodes of an element.

Record Format: Records Written for Any Element File Output Request

Record key	Record type	Attributes
1	Element header record	**1.** Element number or the node number if the subsequent records contain nodal averaged element values. **2.** Integration point number if the subsequent records contain integration point data. Node number if the subsequent records contain data at the nodes of the element. Integration plane number if the subsequent records contain centroidal values for CAXA and SAXA elements. 0 if the subsequent records contain centroidal values or nodal averaged values. **3.** Section point number if this is a shell, beam, or layered solid element and the subsequent records contain data at a section point through the thickness. 0 for continuum elements and for section values in beams and shell elements. **4.** Location identification. 0 if the subsequent records contain data at an integration point; 1 if the subsequent records contain values at the centroid of the element; 2 if the subsequent records contain data at the nodes of the element; 3 if the subsequent records contain data associated with rebar within an element; 4 if the subsequent records contain nodal averaged values; 5 if the subsequent records contain values associated with the whole element. **5.** Rebar name if the subsequent records contain values associated with a named rebar. **6.** Number of direct stresses at a point (`NDI`). **7.** Number of shear stresses at a point (`NSHR`). **8.** 0, currently not used in Abaqus/Standard; number of directions in which displacement or temperature gradients are computed in the element (`NDIR`) in Abaqus/Explicit.

		9. Number of section force or section strain components (`NSFC`).
2	Temperature Output variable: TEMP	**1.** Temperature.
3(S)	Distributed load Output variable: LOADS	**1.** Load type. **2.** Magnitude.
4(S)	Distributed flux Output variable: FLUXS	**1.** Flux type. **2.** Magnitude.
5	Solution-dependent state variables Output variable: SDV	**1.** State variable 1. **2.** State variable 2. **3.** Etc. The record can have up to 80 words in ASCII format or 512 words in binary format. Repeat this record as often as necessary to output all active state variables in the model.
6(S)	Void ratio Output variable: VOIDR	**1.** Void ratio.
7(S)	Foundation pressure Output variable: FOUND	**1.** Foundation type. **2.** Magnitude.
8(S)	Coordinates Output variable: COORD	**1.** First coordinate. **2.** Etc.
9(S)	Field variables Output variable: FV	**1.** First field variable. **2.** Etc.
10(S)	Nodal flux caused by heat Output variable: NFLUX	**1.** Node number. **2.** First flux component. **3.** Etc.
11	Stresses Output variable: S	**1.** First stress component. **2.** Second stress component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
475(S)	Average contact pressure (for link and three-dimensional line gasket elements) Output variable: CS11	**1.** Magnitude (available only when the gasket contact area is specified; see *Defining the Contact Area for Average Contact Pressure Output*).

12(S)	Stress invariants Output variable: SINV	**1.** Mises stress. **2.** Tresca stress. **3.** Hydrostatic pressure. **4.** Currently not used. **5.** Currently not used. **6.** Currently not used. **7.** Third stress invariant.
13	Section forces and moments Output variable: SF	**1.** First section force. **2.** Second section force. **3.** Etc. (See *Abaqus Elements Guide* for a description of which section forces are available for each beam or shell element type.)
449(S)	Effective axial section force Output variable: ESF1	**1.** Effective axial section force for beams and pipes subjected to pressure loading.
14(S)	Energy densities Output variable: ENER	**1.** Strain energy. Elastic strain energy is the only energy density request available in eigenvalue extractions. None of the energy densities are available in modal procedures or direct-solution steady-state dynamics analyses. **2.** Plastic dissipation. **3.** Creep dissipation. **4.** Viscous dissipation. **5.** Electrostatic energy. **6.** Energy dissipated due to electrical conduction. **7.** Damage dissipation.
14(E)	Energy densities Output variable: ENER	**1.** Elastic strain energy. **2.** Plastic dissipation. **3.** Viscoelastic dissipation (not supported for hyperelastic and hyperfoam material models). **4.** Viscous dissipation. **5.** Currently not used. **6.** Currently not used. **7.** Damage dissipation.
15(S)	Nodal forces caused by stress Output variable: NFORC	**1.** Node number. **2.** First force component. **3.** Etc.
16(S)	Maximum section stresses	**1.** Maximum stress on section.

17(S)	*J*s, *K* for LS3S line springs Output variable: JK	**1.** *J* (*J*-integral). **2.** *K* (stress intensity). **3.** J^{el} (elastic part of *J*-integral). **4.** J^{pl} (plastic part of *J*-integral).
17(S)	*J*s, *K*s for LS6 line springs Output variable: JK	**1.** *J* (*J*-integral). **2.** J^{el} (elastic part of *J*-integral). **3.** J^{pl} (plastic part of *J*-integral). **4.** K_I (Mode I stress intensity factor). **5.** K_{II} (Mode II stress intensity factor). **6.** K_{III} (Mode III stress intensity factor).
18(S)	Pore or acoustic pressure Output variable: POR	**1.** Liquid pressure.
19(S)	Energy summed over element Output variable: ELEN	**1.** Kinetic energy. **2.** Strain energy. Elastic strain energy is the only whole element energy request available in eigenvalue extractions. None of the element energies are available in modal procedures or direct-solution steady-state dynamics analyses. **3.** Plastic dissipation. **4.** Creep dissipation. **5.** Viscous dissipation, not including dissipation due to stabilization. **6.** Static dissipation (due to stabilization). **7.** Artificial strain energy. **8.** Electrostatic energy. **9.** Electrical energy dissipated in a conductor. **10.** Damage dissipation.
19(E)	Energy summed over element Output variable: ELEN	**1.** Currently not used. **2.** Strain energy. **3.** Plastic dissipation. **4.** Viscoelastic dissipation (not supported for hyperelastic and hyperfoam material models). **5.** Viscous dissipation. **6.** Artificial strain energy. **7.** Distortion control dissipation. **8.** Currently not used. **9.** Internal heat energy. **10.** Damage dissipation.

21	Total strain in Abaqus/Standard; infinitesimal strain in Abaqus/Explicit Output variable: E	**1.** First strain component. **2.** Second strain component. **3.** Etc. (See *Abaqus Elements Guide* for a definition of the components for a given element type.)
22	Plastic strains Output variable: PE	**1.** First plastic strain component. **2.** Second plastic strain component. **3.** Etc; followed by the equivalent plastic strain, actively yielding flag (yes or no, A8 format), and magnitude of plastic strain in Abaqus/Standard; followed by "0.0, UNUSED, 0.0" in Abaqus/Explicit for consistency with the length of the Abaqus/Standard record. (See *Abaqus Elements Guide* for a definition of the components for a given element type.)
23$^{(S)}$	Creep strains (including swelling) Output variable: CE	**1.** First creep strain component. **2.** Second creep strain component. **3.** Etc; followed by the equivalent creep strain, volumetric swelling strain, and magnitude of creep strain.
24$^{(S)}$	Total inelastic strains Output variable: IE	**1.** First inelastic strain component. **2.** Second inelastic strain component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
25$^{(S)}$	Total elastic strains Output variable: EE	**1.** First elastic strain component. **2.** Second elastic strain component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
26	Unit normal to crack in concrete Output variable: CRACK	**1.** 11-component (if a 1D, 2D, or 3D analysis). **2.** 12-component (if a 2D or 3D analysis). **3.** 13-component (if a 3D analysis). **4.** 21-component (if a 2D or 3D analysis). **5.** 22-component (if a 2D or 3D analysis). **6.** 23-component (if a 3D analysis). **7.** 31-component (if a 3D analysis). **8.** 32-component (if a 3D analysis). **9.** 33-component (if a 3D analysis).

27	Section thickness Output variable: STH	**1.** Current section thickness for membranes and finite-strain shells in Abaqus/Standard and for plane stress elements, membranes, and all shells in Abaqus/Explicit.
28	Heat flux vector Output variable: HFL	**1.** Magnitude. **2.** First component. **3.** Second component. **4.** Etc.
625(S)	Temperature gradient vector Output variable: GRADT	**1.** First component. **2.** Second component. **3.** Etc.
29	Section strains and curvatures Output variable: SE	**1.** First section strain. **2.** Second section strain. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of what section strains are available for each beam or shell element type.)
30(S)	Deformation gradient Output variable: DG	**1.** F_{11}. **2.** Etc. The record will have `NDI` diagonal components of $\mathbf{F}$, then `NSHR` above diagonal components (F_{12}, F_{13}, F_{23}), then `NSHR` below diagonal components (F_{21}, F_{31}, F_{32}), where `NDI` and `NSHR` are given in the element header record (record key 1). Available only for hyperelasticity, hyperfoam, and material models defined in user subroutine *UMAT*.
31(S)	Concrete failure Output variable: CONF	**1.** Summary of the state of a concrete material point. This is the number of cracks or -1 if the concrete has crushed.
32(S)	Strain jumps at nodes Output variable: SJP	**1.** First strain jump component. **2.** Second strain jump component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
33(S)	Film Output variable: FILM	**1.** Type. **2.** Sink temperature. **3.** Film coefficient.

34(S)	Radiation Output variable: RAD	**1.** Type. **2.** Sink temperature. **3.** Radiation constant.
35(S)	Saturation (pore pressure analysis) Output variable: SAT	**1.** Saturation.
36(S)	Substresses (for ITT elements) Output variable: SS	**1.** First substress. **2.** Second substress.
38(S)	Mass concentration (mass diffusion analysis) Output variable: CONC	**1.** Concentration.
446(S)	Amount of solute at the integration point (mass diffusion analysis) Output variable: ISOL	**1.** Amount of solute.
447(S)	Amount of solute in the current element (mass diffusion analysis) Output variable: ESOL	**1.** Amount of solute.
448(S)	Amount of solute in the element set or model (mass diffusion analysis) Output variable: SOL	**1.** Amount of solute.
39(S)	Mass concentration flux vector Output variable: MFL	**1.** Magnitude. **2.** First component. **3.** Second component. **4.** Etc.
39(S)	Mass flow rate Output variable: MFL	**1.** Current flow rate.
40(S)	Gel (pore pressure analysis) Output variable: GELVR	**1.** Gel volume ratio.
43(S)	Total fluid volume ratio Output variable: FLUVR	**1.** Total fluid volume ratio.
61(E)	Element status Output variable: STATUS	**1.** Status of element (shear failure model, tensile failure model, porous failure criterion, brittle failure model, Johnson-Cook plasticity model, and *VUMAT*). The status of an element is 1.0 if the element is active, 0.0 if the element is not.
73(E)	Equivalent plastic strain Output variable: PEEQ	**1.** Equivalent plastic strain. For crushable foam plasticity with volumetric hardening, it is the volumetric compacting plastic

		strain. For cap plasticity it is p_b (the cap position).
74(E)	Mean pressure stress Output variable: PRESS	**1.** Mean pressure stress.
75(E)	Mises equivalent stress Output variable: MISES	**1.** Mises stress.
79(S)	Creep strain rate ratio Output variable: RATIO	**1.** Current maximum ratio of creep strain rate and target creep strain rate.
79(E)	Volumetric strain rate Output variable: ERV	**1.** Volumetric strain rate.
80(S)	Solution-dependent amplitude value Output variable: AMPCU	**1.** Current value of the solution-dependent amplitude.
83(S)	Average shell section stresses Output variable: SSAVG	**1.** First section stress. **2.** Second section stress. **3.** Etc. (See *Abaqus Elements Guide* for a description of which section stresses are available for each shell element type.)
85	Local coordinate directions	**1.** First component of the first direction. **2.** Second component of the first direction. **3.** Third component of the first direction. **4.** First component of the second direction. **5.** Second component of the second direction. **6.** Third component of the second direction.
86	Backstress for kinematic hardening plasticity Output variable: ALPHA	**1.** First $\boldsymbol{\alpha}$ component. **2.** Second $\boldsymbol{\alpha}$ component. **3.** Etc. (The number of components is equal to the number of stress components; see the element description in *Abaqus Elements Guide*.)
87(S)	User-defined output variables Output variable: UVARM	**1.** Output variable 1. **2.** Output variable 2. **3.** Etc.
88(S)	Thermal strains Output variable: THE	**1.** First thermal strain component. **2.** Second thermal strain component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)

89	Logarithmic strains Output variable: LE	**1.** First logarithmic strain component. **2.** Second logarithmic strain component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
90	Nominal strains Output variable: NE	**1.** First nominal strain component. **2.** Second nominal strain component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
91(S)	Mechanical strain rates Output variable: ER	**1.** First strain rate component. **2.** Second strain rate component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
96(S)	Total mass flow through fluid link Output variable: MFLT	**1.** Magnitude.
97(S)	Pore fluid effective velocity vector Output variable: FLVEL	**1.** Magnitude. **2.** First component. **3.** Second component. **4.** Etc.
476(E)	Scaling factor Output variable: EMSF	**1.** Element mass scaling factor.
477(E)	Element time increment Output variable: EDT	**1.** Element stable time increment.

Principal Value Records

Record key	Record type	Attributes

For all principal values, the number of components equals `NDI` unless `NDI` equals 1, in which case the number of components equals `NDI` plus `NSHR`, where `NDI` and `NSHR` are given on the element header record. In the cases where `NDI` equals 2, only the in-plane values are given.

401	Principal stresses Output variable: SP	**1.** Minimum principal stress. **2.** Etc.

402	Principal values of backstress tensor for kinematic hardening plasticity Output variable: ALPHAP	**1.** Minimum principal value. **2.** Etc.
403	Principal strains Output variable: EP	**1.** Minimum principal strain. **2.** Etc.
404	Principal nominal strains Output variable: NEP	**1.** Minimum principal nominal strain. **2.** Etc.
405	Principal logarithmic strains Output variable: LEP	**1.** Minimum principal logarithmic strain. **2.** Etc.
406(S)	Principal mechanical strain rates Output variable: ERP	**1.** Minimum principal strain rate. **2.** Etc.
407(S)	Principal values of deformation gradient Output variable: DGP	**1.** Minimum principal value. **2.** Etc.
408(S)	Principal elastic strains Output variable: EEP	**1.** Minimum principal elastic strain. **2.** Etc.
409(S)	Principal inelastic strains Output variable: IEP	**1.** Minimum principal inelastic strain. **2.** Etc.
410(S)	Principal thermal strains Output variable: THEP	**1.** Minimum principal thermal strain. **2.** Etc.
411(S)	Principal plastic strains Output variable: PEP	**1.** Minimum principal plastic strain. **2.** Etc.
412(S)	Principal creep strains Output variable: CEP	**1.** Minimum principal creep strain. **2.** Etc.

Records for Porous Metal Plasticity

Record key	Record type	Attributes
413	Void volume fraction Output variable: VVF	**1.** f.
414	Void volume fraction (growth) Output variable: VVFG	**1.** f_{gr}.

415	Void volume fraction (nucleation) Output variable: VVFN	**1.** f_{nucl}.
416(S)	Relative density Output variable: RD	**1.** $r = 1 - f$

Records for Brittle Cracking

Record key	Record type	Attributes
421(E)	Cracking strains Output variable: CKE	**1.** First cracking strain component. **2.** Second cracking strain component. **3.** Etc. (See *Abaqus Elements Guide* for a definition of the number and the type of the components for the element type.)
422(E)	Local cracking strains Output variable: CKLE	**1.** First strain component in local crack directions. **2.** Second strain component in local crack directions. **3.** Etc. (See *Abaqus Elements Guide* for a definition of the number and the type of the components for the element type.)
423(E)	Local cracking stresses Output variable: CKLS	**1.** First stress component in local crack directions. **2.** Second stress component in local crack directions. **3.** Etc. (See *Abaqus Elements Guide* for a definition of the number and the type of the components for the element type.)
424(E)	Status of cracks Output variable: CKSTAT	**1.** Status of first crack (if a 1D, 2D, or 3D analysis). CKSTAT can have the following values: 0.0=uncracked, 1.0=closed crack, 2.0=actively cracking, 3.0=crack closing/reopening. **2.** Status of second crack (if a 2D or 3D analysis). **3.** Status of third crack (if a 3D analysis).
441(E)	Cracking strain magnitude Output variable: CKEMAG	**1.** Magnitude of cracking strain.

Records for Inelastic Nonlinear Response in a Beam General Section

Record key	Record type	Attributes
42(S)	Plastic strain components Output variable: SPE	**1.** Axial plastic strain. **2.** Curvature change about the local 1-axis. **3.** Curvature change about the local 2-axis (available only for 3D beams). **4.** Twist of the beam (available only for 3D beams).
47(S)	Equivalent plastic strains Output variable: SEPE	**1.** Axial equivalent plastic strain. **2.** Curvature change about the local 1-axis. **3.** Curvature change about the local 2-axis (available only for 3D beams). **4.** Twist of the beam (available only for 3D beams).

Records for Elastic-Plastic Response in Frame Elements

Record key	Record type	Attributes
462(S)	Elastic section strain components Output variable: SEE	**1.** Elastic axial strain. **2.** Elastic curvature change about the local 1-axis. **3.** Elastic curvature change about the local 2-axis (available only for 3D frame elements). **4.** Elastic twist of the beam (available only for 3D frame elements).
463(S)	Plastic displacements at frame element's ends Output variable: SEP	**1.** Plastic axial displacement. **2.** Plastic rotation about the local 1-axis. **3.** Plastic rotation about the local 2-axis (available only for 3D frame elements). **4.** Plastic rotation about the element axis (available only for 3D frame elements). **5.** Actively yielding flag (yes or no, A8 format) for frame element's end sections. **6.** Buckling flag (yes, no, or na; A8 format) for frame element's end sections.
464(S)	Generalized backstress components Output variable: SALPHA	**1.** Axial backstress component. **2.** Bending backstress about the local 1-axis.

		3. Bending backstress about the local 2-axis (available only for 3D frame elements). **4.** Twist backstress of the beam (available only for 3D frame elements).

Records for Connector Elements

Record key	Record type	Attributes
495	Connector total force Output variable: CTF	**1.** First component of total force. **2.** Second component of total force. **3.** Etc.
496	Connector elastic force Output variable: CEF	**1.** First component of elastic force. **2.** Second component of elastic force. **3.** Etc.
497	Connector viscous force Output variable: CVF	**1.** First component of viscous force. **2.** Second component of viscous force. **3.** Etc.
498	Connector friction force Output variable: CSF	**1.** First component of friction force. **2.** Second component of friction force. **3.** Etc.
499	Connector lock and connector stop status flags Output variable: CSLST	**1.** Flag in the 1-direction. **2.** Flag in the 2-direction. **3.** Etc.
500	Connector reaction force Output variable: CRF	**1.** First component of reaction force. **2.** Second component of reaction force. **3.** Etc.
501	Connector concentrated force Output variable: CCF	**1.** First component of concentrated force. **2.** Second component of concentrated force. **3.** Etc.
502	Connector relative position Output variable: CP	**1.** First component of relative position. **2.** Second component of relative position. **3.** Etc.
503	Connector relative displacement Output variable: CU	**1.** First component of relative displacement. **2.** Second component of relative displacement.

		3. Etc.
504	Connector constitutive displacement Output variable: CCU	**1.** First component of constitutive displacement. **2.** Second component of constitutive displacement. **3.** Etc.
505	Connector relative velocity Output variable: CV	**1.** First component of relative velocity. **2.** Second component of relative velocity. **3.** Etc.
506	Connector relative acceleration Output variable: CA	**1.** First component of relative acceleration. **2.** Second component of relative acceleration. **3.** Etc.
507[(E)]	Connector failure status flags Output variable: CFAILST	**1.** Flag in the 1-direction. **2.** Flag in the 2-direction. **3.** Etc.
542	Connector friction-generating contact force Output variable: CNF	**1.** First component of friction-generating force. **2.** Second component of friction-generating force. **3.** Etc.
546	Connector relative velocity in the direction of instantaneous slip Output variable: CIVC	**1.** Relative velocity in the direction of instantaneous slip.
548	Accumulated frictional slip Output variable: CASU	**1.** First component of accumulated frictional slip. **2.** Second component of accumulated frictional slip. **3.** Etc.
556	Connector elastic displacement Output variable: CUE	**1.** First component of elastic displacement. **2.** Second component of elastic displacement. **3.** Etc.
557	Connector plastic relative displacement Output variable: CUP	**1.** First component of plastic relative displacement. **2.** Second component of plastic relative displacement. **3.** Etc.

558	Connector equivalent plastic relative displacement Output variable: CUPEQ	**1.** First component of equivalent plastic relative displacement. **2.** Second component of equivalent plastic relative displacement. **3.** Etc.
559(E)	Connector overall damage variable Output variable: CDMG	**1.** First component of overall damage variable. **2.** Second component of overall damage variable. **3.** Etc.
560(E)	Connector force-based damage initiation criterion Output variable: CDIF	**1.** First component of connector force-based damage initiation criterion. **2.** Second component of connector force-based damage initiation criterion. **3.** Etc.
561(E)	Connector motion-based damage initiation criterion Output variable: CDIM	**1.** First component of connector motion-based damage initiation criterion. **2.** Second component of connector motion-based damage initiation criterion. **3.** Etc.
562(E)	Connector plastic motion-based damage initiation criterion Output variable: CDIP	**1.** First component of connector plastic motion-based damage initiation criterion. **2.** Second component of connector plastic motion-based damage initiation criterion. **3.** Etc.
563	Connector kinematic hardening shift force Output variable: CALPHAF	**1.** First component of connector kinematic hardening shift force. **2.** Second component of connector kinematic hardening shift force. **3.** Etc.

Record for Plane Stress Orthotropic Failure Measures

Record key	Record type	Attributes
44(S)	Failure measures Output variable: CFAILURE	**1.** Maximum stress theory. **2.** Tsai-Hill theory. **3.** Tsai-Wu theory. **4.** Azzi-Tsai-Hill theory. **5.** Maximum strain theory.

Record for Equivalent Plastic Strain Components for Cap Plasticity

Record key	Record type	Attributes
45	Equivalent plastic strain components Output variable: PEQC	**1.** Equivalent plastic strain for Drucker-Prager failure surface. **2.** Actively yielding flag (yes or no, A8 format) for Drucker-Prager failure surface. **3.** Equivalent plastic strain for cap surface. **4.** Actively yielding flag (yes or no, A8 format) for cap surface. **5.** Equivalent plastic strain for transition surface. **6.** Actively yielding flag (yes or no, A8 format) for transition surface. **7.** Total volumetric inelastic strain. **8.** Actively yielding flag (yes or no, A8 format).

Record for Equivalent Plastic Strain Components for Jointed Materials

Record key	Record type	Attributes
45$^{(S)}$	Equivalent plastic strain components Output variable: PEQC	**1.** Equivalent plastic strain for joint 1. **2.** Actively yielding flag (yes or no, A8 format) for joint 1. **3.** Equivalent plastic strain for joint 2. **4.** Actively yielding flag (yes or no, A8 format) for joint 2. **5.** Equivalent plastic strain for joint 3. **6.** Actively yielding flag (yes or no, A8 format) for joint 3. **7.** Equivalent plastic strain for bulk material. **8.** Actively yielding flag (yes or no, A8 format) for bulk material.

Record for Equivalent Plastic Strain in Uniaxial Tension for Cast Iron Plasticity

Record key	Record type	Attributes
473$^{(S)}$	Equivalent plastic strain in uniaxial tension Output variable: PEEQT	**1.** Equivalent plastic strain in uniaxial tension for cast iron plasticity model. **2.** Actively yielding flag (yes or no, A8 format).

Records for Two-Layer Viscoplasticity

Record key	Record type	Attributes
$22^{(S)}$	Plastic strains in the elastic-plastic network Output variable: PE	**1.** First plastic strain component. **2.** Second plastic strain component. **3.** Etc.; followed by the equivalent plastic strain, actively yielding flag (yes or no, A8 format), and magnitude of plastic strain. (See *Abaqus Elements Guide* for a definition of the components for a given element type.)
$524^{(S)}$	Stresses in the elastic-viscous network Output variable: VS	**1.** First stress component. **2.** Second stress component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
$525^{(S)}$	Stresses in the elastic-plastic network Output variable: PS	**1.** First stress component. **2.** Second stress component. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
$526^{(S)}$	Viscous strains in the elastic-viscous network Output variable: VE	**1.** First viscous strain component. **2.** Second viscous strain component. **3.** Etc.; followed by the equivalent viscous strain.

Record for Elements with Electric Potential Degrees of Freedom

Record key	Record type	Attributes
$50^{(S)}$	Electrical potential gradients Output variable: EPG	**1.** Magnitude. **2.** First potential gradient. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)

Records for Rebar Quantities

Record key	Record type	Attributes
442	Force in rebar Output variable: RBFOR	**1.** Magnitude.
443	Rebar angle Output variable: RBANG	**1.** Angle in degrees between the reinforcing and the user-specified isoparametric direction. Available only for membrane, shell, and surface elements.
444	Change in rebar angle Output variable: RBROT	**1.** Change in angle in degrees between the reinforcing and the user-specified isoparametric direction. Available only for membrane, shell, and surface elements.

Record for Forced Convection/Diffusion Heat Transfer Elements

Record key	Record type	Attributes
445$^{(S)}$	Mass flow rates Output variable: MFR	**1.** First mass flow rate. **2.** Etc.

Records for Piezoelectric Materials

Record key	Record type	Attributes
46$^{(S)}$	Magnitudes and phases of potential gradients (linear dynamics only) Output variable: PHEPG	**1.** Magnitude of first electrical potential gradient. **2.** Magnitude of second electrical potential gradient. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.) **4.** Phase angle of first electrical potential gradient. **5.** Phase angle of second electrical potential gradient. **6.** Etc.
49$^{(S)}$	Magnitudes and phases of electrical charge fluxes (linear dynamics only)	**1.** Magnitude of first charge flux.

	Output variable: PHEFL	**2.** Magnitude of second charge flux. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.) **4.** Phase angle of first charge flux. **5.** Phase angle of second charge flux. **6.** Etc.
51$^{(S)}$	Electrical charge fluxes Output variable: EFLX	**1.** Magnitude. **2.** First charge flux. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
60$^{(S)}$	Distributed electrical charges Output variable: CHRGS	**1.** Charge type. **2.** Magnitude.

Records for Coupled Thermal-Electric Elements

Record key	Record type	Attributes
425$^{(S)}$	Electrical current density Output variable: ECD	**1.** Magnitude. **2.** First current density. **3.** Etc. (See the element description in *Abaqus Elements Guide* for a definition of the number and type of the components for the element type.)
426$^{(S)}$	Distributed electrical current density Output variable: ECURS	**1.** Electrical current type. **2.** Magnitude.
427$^{(S)}$	Nodal current due to electric conduction Output variable: NCURS	**1.** Node number. **2.** Magnitude.

Records for Cohesive Elements

Record key	Record type	Attributes
252$^{(S)}$	All active components of the damage initiation criteria Output variable: DMICRT	**1.** MAXSCRT, maximum nominal stress damage initiation criterion. **2.** MAXECRT, maximum nominal strain damage initiation criterion.

Record key	Record type	Attributes
		3. QUADSCRT, quadratic nominal stress damage initiation criterion. **4.** QUADECRT, quadratic nominal strain damage initiation criterion.
235(S)	Overall scalar stiffness degradation Output variable: SDEG	**1.** Magnitude.
61(S)	Element status Output variable: STATUS	**1.** Status of the element (the status of an element is 1.0 if the element is active, 0.0 if the element is not).

Records for Equivalent Rigid Body Variables in Direct-Integration Implicit Dynamic Analyses

Record key	Record type	Attributes

Records 52–59 provide values summed over an element set. These variables are available only in direct-integration implicit dynamic analyses (see *Implicit Dynamic Analysis Using Direct Integration*).

Record key	Record type	Attributes
52(S)	Current coordinates of center of mass Output variable: XC	**1.** Coordinate 1. **2.** Coordinate 2. **3.** Etc. (The number of components depends upon the overall dimensionality of the element set.)
53(S)	Displacement of the center of mass Output variable: UC	**1.** Displacement 1. **2.** Displacement 2. **3.** Etc. (The number of components depends upon the overall dimensionality of the element set.)
54(S)	Equivalent rigid body velocity Output variable: VC	**1.** Component 1. **2.** Component 2. **3.** Etc. (The number of components depends upon the overall dimensionality of the element set.)
55(S)	Angular momentum about center of mass Output variable: HC	**1.** Component 1. **2.** Component 2. **3.** Etc. (The number of components depends upon the overall dimensionality of the element set.)
56(S)	Angular momentum about origin Output variable: HO	**1.** Component 1. **2.** Component 2.

Record key	Record type	Attributes
		3. Etc. (The number of components depends upon the overall dimensionality of the element set.)
57(S)	Rotary inertia about the origin Output variable: RI	**1.** Component 11. **2.** Component 22. **3.** Etc. (The number of components depends upon the overall dimensionality of the element set.)
58(S)	Current mass of element set Output variable: MASS	**1.** Mass.
59(S)	Current volume of element set Output variable: VOL	**1.** Volume. (Only available for continuum and structural elements not using general beam or shell section definitions.)

Record for Transverse Shear Stress in Thick Shell Elements Such as S3R, S4R, S8R, and S8RT

Record key	Record type	Attributes
48	Transverse shear stresses in 13 and 23 planes Output variable: TSHR	**1.** Component 13. **2.** Component 23.

Records for Linear Dynamics

Record key	Record type	Attributes
62(S)	Magnitude and phase angle of stress components Output variable: PHS	**1.** Magnitude of first stress component. **2.** Magnutude of second stress component. **3.** Etc. **4.** Phase angle of first stress component. **5.** Phase angle of second stress component. **6.** Etc.
63(S)	RMS values of stress components Output variable: RS	**1.** First component of stress. **2.** Second component of stress. **3.** Etc.
65(S)	Magnitude and phase angle of strain components Output variable: PHE	**1.** Magnitude of first strain component. **2.** Magnitude of second strain component. **3.** Etc. **4.** Phase angle of first strain component. **5.** Phase angle of second strain component.

		6. Etc.
66(S)	RMS values of strain components Output variable: RE	**1.** First component of strain. **2.** Second component of strain. **3.** Etc.

Records for Connector Elements (Available Only for Linear Dynamics)

Record key	Record type	Attributes
508(S)	Magnitude and phase angle of connector total forces Output variable: PHCTF	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.
509(S)	Magnitude and phase angle of connector elastic forces Output variable: PHCEF	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.
510(S)	Magnitude and phase angle of connector viscous forces Output variable: PHCVF	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.
511(S)	Magnitude and phase angle of connector reaction forces Output variable: PHCRF	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.
520(S)	Magnitude and phase angle of connector friction forces Output variable: PHCSF	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.

512$^{(S)}$	Magnitude and phase angle of connector relative displacements Output variable: PHCU	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.
513$^{(S)}$	Magnitude and phase angle of connector constitutive displacements Output variable: PHCCU	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.
522$^{(S)}$	Magnitude and phase angle of connector relative velocities Output variable: PHCV	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.
523$^{(S)}$	Magnitude and phase angle of connector relative accelerations Output variable: PHCA	**1.** Magnitude of the first component. **2.** Magnitude of the second component. **3.** Etc. **4.** Phase angle of the first component. **5.** Phase angle of the second component. **6.** Etc.
543$^{(S)}$	Magnitude and phase angle of friction-generating connector force Output variable: PHCNF	**1.** Magnitude of the first component of friction-generating connector force. **2.** Magnitude of the second component of friction-generating connector force. **3.** Etc. **4.** Phase angle of the first component of friction-generating connector force. **5.** Phase angle of the second component of friction-generating connector force. **6.** Etc.
547$^{(S)}$	Magnitude and phase angle of connector relative velocity in the direction of instantaneous slip Output variable: PHCIVSL	**1.** Magnitude of connector relative velocity in the direction of instantaneous slip. **2.** Phase angle of connector relative velocity in the direction of instantaneous slip.
514$^{(S)}$	RMS values of connector total forces Output variable: RCTF	**1.** First component of force. **2.** Second component of force. **3.** Etc.

515(S)	RMS values of connector elastic forces Output variable: RCEF	**1.** First component of force. **2.** Second component of force. **3.** Etc.
516(S)	RMS values of connector viscous forces Output variable: RCVF	**1.** First component of force. **2.** Second component of force. **3.** Etc.
517(S)	RMS values of connector reaction forces Output variable: RCRF	**1.** First component of force. **2.** Second component of force. **3.** Etc.
521(S)	RMS values of connector friction forces Output variable: RCSF	**1.** First component of force. **2.** Second component of force. **3.** Etc.
518(S)	RMS values of connector relative displacements Output variable: RCU	**1.** First component of relative displacements. **2.** Second component of relative displacements. **3.** Etc.
519(S)	RMS values of connector constitutive displacements Output variable: RCCU	**1.** First component of constitutive displacements. **2.** Second component of constitutive displacements. **3.** Etc.
544(S)	RMS values of connector force generating friction Output variable: RCNF	**1.** RMS values of first component of friction-generating connector force. **2.** RMS values of second component of friction-generating connector force. **3.** Etc.

Records for Fluid Link Elements (Available Only for Linear Dynamics)

Record key	Record type	Attributes
94(S)	Magnitude and phase angle of mass flow rate Output variable: PHMFL	**1.** Magnitude. **2.** Phase angle.
95(S)	Magnitude and phase angle of total mass flow Output variable: PHMFT	**1.** Magnitude. **2.** Phase angle.

Records for Output of Element Volumes

Record key	Record type	Attributes

The following three variables are not available for eigenfrequency extraction, complex eigenfrequency extraction, eigenvalue buckling prediction, or linear dynamics procedures. They are available only for continuum and structural elements not using general beam or shell section definitions.

Record key	Record type	Attributes
76(S)	Integration point volume Output variable: IVOL	**1.** Current integration point volume. Section point volume in the case of beams and shells.
77(S)	Section volume Output variable: SVOL	**1.** Current section volume.
78(S)	Whole element volume Output variable: EVOL	**1.** Current element volume.

Record for Solid Elements in an Adaptive Mesh Domain in Abaqus/Standard

Record key	Record type	Attributes
264(S)	Change in volume. Output variable: VOLC	**1.** Change in area or volume of an element set solely due to adaptive meshing.

Records Written for Any Node File Output Request

This section describes the format of the individual records in the Abaqus results file.

Record Format: Records Written for Any Node File Output Request

Record key	Record type	Attributes
101	Displacements Output variable: U	**1.** Node number. **2.** First component of displacement. **3.** Second component of displacement. **4.** Etc.
102	Velocities Output variable: V	**1.** Node number. **2.** First component of velocity. **3.** Second component of velocity. **4.** Etc.
103	Accelerations Output variable: A	**1.** Node number. **2.** First component of acceleration. **3.** Second component of acceleration. **4.** Etc.
104	Reaction forces Output variable: RF	**1.** Node number. **2.** First component of reaction force. **3.** Second component of reaction force. **4.** Etc.
105$^{(S)}$	Electrical potential Output variable: EPOT	**1.** Node number. **2.** Magnitude.
106$^{(S)}$	Point loads, moments, fluxes Output variable: CF	**1.** Node number. **2.** First component of load or flux. **3.** Second component of load or flux. **4.** Etc.
107	Coordinates Output variable: COORD	**1.** Node number. **2.** First coordinate. **3.** Second coordinate. **4.** Etc.
108	Pore or acoustic pressure Output variable: POR	**1.** Node number. **2.** Pressure.
109$^{(S)}$	Reactive fluid volume flux	**1.** Node number.

	Output variable: RVF	**2.** Reaction fluid volume flux.
110(S)	Reactive fluid total volume Output variable: RVT	**1.** Node number. **2.** Reaction fluid total volume.
119(S)	Electrical reaction charges Output variable: RCHG	**1.** Node number. **2.** Charge scalar value.
120(S)	Concentrated electrical nodal charges Output variable: CECHG	**1.** Node number. **2.** Current scalar value.
136	Fluid cavity pressure Output variable: PCAV	**1.** Fluid cavity reference node number. **2.** Pressure.
137	Fluid cavity volume Output variable: CVOL	**1.** Fluid cavity reference node number. **2.** Volume.
138(S)	Electrical reaction current Output variable: RECUR	**1.** Node number. **2.** Electrical current.
139(S)	Concentrated electrical nodal current Output variable: CECUR	**1.** Node number. **2.** Electrical current.
145(S)	Viscous forces due to static stabilization Output variable: VF	**1.** Node number. **2.** First component of viscous force. **3.** Second component of viscous force. **4.** Etc.
146(S)	Total forces Output variable: TF	**1.** Node number. **2.** First component of total force. **3.** Second component of total force. **4.** Etc.
151(E)	Acoustic absolute pressure Output variable: PABS	**1.** Node number. **2.** Absolute pressure.
201	Temperatures Output variable: NT	**1.** Node number. **2.** Temperature. **3.** Etc. (for heat shells)
204(S)	Residual fluxes Output variable: RFL	**1.** Node number. **2.** Residual flux. **3.** Etc. (for heat shells)

Record key	Record type	Attributes
204(E)	Reaction fluxes Output variable: RFL	**1.** Node number. **2.** First component of reaction flux. **3.** Second component of reaction flux. **4.** Etc.
206(S)	Concentrated fluxes Output variable: CFL	**1.** Node number. **2.** Concentrated flux. **3.** Etc. (for heat shells)
214(S)	Internal fluxes Output variable: RFLE	**1.** Node number. **2.** Flux, excluding external flux. **3.** Etc. (for heat shells)
221(S)	Normalized concentration (mass diffusion analysis) Output variable: NNC	**1.** Node number. **2.** Concentration.
237(S)	Motions (in cavity radiation analysis) Output variable: MOT	**1.** Node number. **2.** First component of motion. **3.** Second component of motion. **4.** Etc.
320(S)	Concentrated fluid flow Output variable: CFF	**1.** Node number. **2.** Magnitude of fluid flow.

Records for Linear Dynamics

Record key	Record type	Attributes
111(S)	Magnitude and phase angle of relative displacement Output variable: PU	**1.** Node number. **2.** Magnitude of first displacement component. **3.** Magnitude of second displacement component. **4.** Etc. **5.** Phase angle of first displacement component. **6.** Phase angle of second displacement component. **7.** Etc.
112(S)	Magnitude and phase angle of total displacement Output variable: PTU	**1.** Node number. **2.** Magnitude of first displacement component.

Record type	Description	Attributes
		3. Magnitude of second displacement component. **4.** Etc. **5.** Phase angle of first displacement component. **6.** Phase angle of second displacement component. **7.** Etc.
113(S)	Total displacement Output variable: TU	**1.** Node number. **2.** First component of displacement. **3.** Second component of displacement. **4.** Etc.
114(S)	Total velocity Output variable: TV	**1.** Node number. **2.** First component of velocity. **3.** Second component of velocity. **4.** Etc.
115(S)	Total acceleration Output variable: TA	**1.** Node number. **2.** First component of acceleration. **3.** Second component of acceleration. **4.** Etc.
116(S)	Magnitude and phase angle of acoustic or fluid cavity pressure Output variable: PPOR	**1.** Node number. **2.** Magnitude of pressure. **3.** Phase angle of pressure.
117(S)	Magnitude and phase angle of electrical potential Output variable: PHPOT	**1.** Node number. **2.** Magnitude of potential. **3.** Phase angle of potential.
118(S)	Magnitude and phase angle of reactive charge (piezoelectric analysis) Output variable: PHCHG	**1.** Node number. **2.** Magnitude of charge. **3.** Phase angle of charge.
123(S)	RMS values of relative displacement Output variable: RU	**1.** Node number. **2.** First component of displacement. **3.** Second component of displacement. **4.** Etc.
124(S)	RMS values of total displacement Output variable: RTU	**1.** Node number. **2.** First component of displacement. **3.** Second component of displacement. **4.** Etc.

127(S)	RMS values of relative velocity Output variable: RV	**1.** Node number. **2.** First component of velocity. **3.** Second component of velocity. **4.** Etc.
128(S)	RMS values of total velocity Output variable: RTV	**1.** Node number. **2.** First component of velocity. **3.** Second component of velocity. **4.** Etc.
131(S)	RMS values of relative acceleration Output variable: RA	**1.** Node number. **2.** First component of acceleration. **3.** Second component of acceleration. **4.** Etc.
132(S)	RMS values of total acceleration Output variable: RTA	**1.** Node number. **2.** First component of acceleration. **3.** Second component of acceleration. **4.** Etc.
134(S)	RMS values of reaction forces Output variable: RRF	**1.** Node number. **2.** First component of reaction force. **3.** Second component of reaction force. **4.** Etc.
135(S)	Magnitude and phase angle of reaction force Output variable: PRF	**1.** Node number. **2.** Magnitude of first component of reaction force. **3.** Magnitude of second component of reaction force. **4.** Etc. **5.** Phase angle of first component of reaction force. **6.** Phase angle of second component of reaction force. **7.** Etc.

Records Written for Any Modal File Output Request during Mode-Based Dynamic Analysis

This section describes the format of the individual records in the Abaqus results file.

Record Format: Records Written for Any Modal File Output Request during Mode-Based Dynamic Analysis

Record key	Record type	Attributes
$301^{(S)}$	Generalized displacements Output variable: GU	**1.** First generalized displacement. **2.** Second generalized displacement. **3.** Etc.
$302^{(S)}$	Generalized velocities Output variable: GV	**1.** First generalized velocity. **2.** Second generalized velocity. **3.** Etc.
$303^{(S)}$	Generalized accelerations Output variable: GA	**1.** First generalized acceleration. **2.** Second generalized acceleration. **3.** Etc.
$304^{(S)}$	Base motions Output variable: BM	**1.** 1 if displacement, 2 if velocity, 3 if acceleration. **2.** x-direction component. **3.** y-direction component. **4.** z-direction component. **5.** x-rotation component. **6.** y-rotation component. **7.** z-rotation component. **8.** Base name.
$305^{(S)}$	Phase angle of generalized displacement Output variable: GPU	**1.** Phase angle of generalized displacement for first mode. **2.** Phase angle of generalized displacement for second mode. **3.** Etc.
$306^{(S)}$	Phase angle of generalized velocity Output variable: GPV	**1.** Phase angle of generalized velocity for first mode. **2.** Phase angle of generalized velocity for second mode. **3.** Etc.
$307^{(S)}$	Phase angle of generalized acceleration Output variable: GPA	**1.** Phase angle of generalized acceleration for first mode.

		2. Phase angle of generalized acceleration for second mode. **3.** Etc.
308(S)	Strain energy per mode Output variable: SNE	**1.** Strain energy for first mode. **2.** Strain energy for second mode. **3.** Etc.
309(S)	Kinetic energy per mode Output variable: KE	**1.** Kinetic energy for first mode. **2.** Kinetic energy for second mode. **3.** Etc.
310(S)	External work per mode Output variable: T	**1.** External work for first mode. **2.** External work for second mode. **3.** Etc.

Records Written for Any Element Matrix or Substructure Matrix File Output Request

This section describes the format of the individual records in the Abaqus results file.

The ordering of variables on element matrices is the same as that used for user elements (see *User-Defined Elements*): first the variables at the element's first node, then those at its second node, etc. Abaqus allows elements to have repeated nodes.

Record Format: Records Written for Any Element Matrix or Substructure Matrix File Output Request

Record key	Record type	Attributes
1001$^{(S)}$	Element matrix header record	**1.** Element number (zero if this is a substructure). **2.** Element or substructure type in A8 format. **3.** Number of nodes on the element. **4.** Node number of the element's first node. **5.** Node number of the element's second node. **6.** Etc.
1002$^{(S)}$	Element or substructure recovery matrix nodal DOF	**1.** First DOF at the element's or at the recovery matrix's first retained node. **2.** Second DOF at the element's or at the recovery matrix's first retained node. **3.** Etc.
1003$^{(S)}$	Element or substructure recovery matrix nodal DOF change	**1.** Node where the DOFs change. **2.** First DOF at this node. **3.** Second DOF at this node. **4.** Etc.
1004$^{(S)}$	Element matrix record size	**1.** Maximum record length (including the record length and record key words) for element matrix and load vector records that follow. The matrix or load vector records will be subdivided into multiple records as needed to fit within this maximum length. The record key for any continuation record will be the same as for the first record.
1005$^{(S)}$	Element matrix header (continued)	**1.** Element node number continued from record 1001 (if necessary). **2.** Etc.
1011$^{(S)}$	Symmetric element stiffness matrix	**1.** (1, 1) stiffness. **2.** (1, 2) stiffness.

		3. (2, 2) stiffness. **4.** Etc., stored in columns, from the first row to the diagonal term of each column.
1012(S)	Nonsymmetric element stiffness matrix	**1.** (1, 1) stiffness. **2.** (2, 1) stiffness. **3.** (3, 1) stiffness. **4.** Etc., stored in columns.
1021(S)	Symmetric element mass matrix	**1.** (1, 1) mass. **2.** (1, 2) mass. **3.** (2, 2) mass. **4.** Etc., stored in columns, from the first row to the diagonal term of each column.
1022(S)	Nonsymmetric element mass matrix	**1.** (1, 1) mass. **2.** (2, 1) mass. **3.** (3, 1) mass. **4.** Etc., stored in columns.
1031(S)	Load vector	**1.** Load case. **2.** Load on first DOF. **3.** Load on second DOF. **4.** Etc.
1032(S)	Substructure load case vector	**1.** Load case name (A8 format). **2.** Load on first DOF. **3.** Load on second DOF. **4.** Etc.
1041(S)	Substructure recovery matrix header record	**1.** Zero. **2.** Element or substructure type in A8 format. **3.** Number of eliminated nodes. **4.** Node number of the first eliminated node. **5.** Node number of the second eliminated node. **6.** Etc.
1042(S)	Substructure recovery matrix	**1.** Column number corresponding to the retained DOFs list. **2.** Coefficient of first eliminated DOF. **3.** Coefficient of second eliminated DOF. **4.** Etc.
1043(S)	Substructure recovery matrix header (continued)	**1.** Node number continued from record 1041 (if necessary).

2. Etc.

Record Written for Any Energy File Output Request

The section describes the format of the individual records in the Abaqus results file.

When you do not specify an element set for which energy output is being requested in Abaqus/Standard, record 1999 provides values summed over the entire model; when you specify an element set for energy output, record 1999 provides values summed over all the elements in the specified element set. You can distinguish between a whole model 1999 energy record and an element set 1999 energy record by searching for a 1911 output request definition record containing the element set name; this 1911 record will be written just before the element set 1999 energy record. This 1911 record also has the first attribute set to 3 to indicate element set output. In Abaqus/Explicit you cannot specify selected element sets for an energy output request; record 1999 provides the total energies for the whole model.

Record Format: Record Written for Any Energy File Output Request

Record key	Record type	Attributes
1999$^{(S)}$	Total energies record	**1.** Total kinetic energy (ALLKE). **2.** Total recoverable (elastic) strain energy (ALLSE). **3.** Total external work (ALLWK, available only for the whole model.) **4.** Total plastic dissipation (ALLPD). **5.** Total creep dissipation (ALLCD). **6.** Total viscous dissipation, not including dissipation due to stabilization (ALLVD). **7.** Total loss of kinetic energy at impacts (ALLKL, available only for the whole model). **8.** Total artificial strain energy (ALLAE). **9.** Total energy dissipated through quiet boundaries (ALLQB, available only for the whole model). **10.** Total electrostatic energy (ALLEE). **11.** Total strain energy (ALLIE). **12.** Total energy balance (ETOTAL, available only for the whole model). **13.** Total energy dissipated through frictional effects (ALLFD, available only for the whole model). **14.** Total electrical energy dissipated in conductors (ALLJD). **15.** Total static dissipation (due to stabilization, ALLSD). **16.** Total damage dissipation (ALLDMD). **17.** Currently not used. **18.** Currently not used.
1999$^{(E)}$	Total energies record	**1.** Total kinetic energy (ALLKE).

2. Total recoverable (elastic) strain energy (ALLSE).
3. Total external work (ALLWK).
4. Total plastic dissipation (ALLPD).
5. Total viscoelastic dissipation (ALLCD).
6. Total viscous dissipation (ALLVD, not supported for hyperelastic and hyperfoam material models).
7. Currently not used.
8. Total artificial strain energy (ALLAE).
9. Total distortion control dissipation energy (ALLDC).
10. Currently not used.
11. Total strain energy (ALLIE).
12. Total energy balance (ETOTAL).
13. Total energy dissipated through frictional effects (ALLFD).
14. Currently not used.
15. Percent change in mass (DMASS).
16. Total damage dissipation (ALLDMD).
17. Internal heat energy (ALLIHE).
18. External heat energy (ALLHF).

Records Written for Contour Integrals

This section describes the format of the individual records in the Abaqus results file.

Calculations of the J-integral and the C_t-integral, the stress intensity factors, the crack propagation direction, and the T-stress can be requested. The record is written for each crack, one record per crack front location. See record key 17 for J-integral values for line spring elements.

Record Format: Records Written for Contour Integrals

Record key	Record type	Attributes
1991(S)	J-integral values	**1.** Crack number. **2.** Node set (A8 format). **3.** Number of contours. **4.** J-integral value estimated by first contour. **5.** J-integral value estimated by second contour. **6.** Etc.
1992(S)	C_t-integral values	**1.** Crack number. **2.** Node set (A8 format). **3.** Number of contours. **4.** C_t-integral value estimated by first contour. **5.** C_t-integral value estimated by second contour. **6.** Etc.
1995(S)	Stress intensity factors	**1.** Crack number. **2.** Node set (A8 format). **3.** Number of contours. **4.** K_I (Mode I stress intensity factor) estimated by first contour. **5.** K_{II} (Mode II stress intensity factor) estimated by first contour. **6.** K_{III} (Mode III stress intensity factor) estimated by first contour (available only for 3D elements). **7.** Crack propagation direction (in degrees) estimated by first contour (available only for homogeneous, isotropic elastic materials). **8.** J-integral value estimated from stress intensity factors of first contour. **9.** K_I (Mode I stress intensity factor) estimated by second contour.

		10. K_{II} (Mode II stress intensity factor) estimated by second contour. **11.** K_{III} (Mode III stress intensity factor) estimated by second contour (available only for 3D elements). **12.** Crack propagation direction (in degrees) estimated by second contour (available only for homogeneous, isotropic elastic materials). **13.** *J*-integral value estimated from stress intensity factors of second contour. **14.** Etc.
1996(S)	*T*-stress values	**1.** Crack number. **2.** Node set (A8 format). **3.** Number of contours. **4.** *T*-stress value estimated by first contour. **5.** *T*-stress value estimated by second contour. **6.** Etc.

Record Written for Crack Propagation Analysis

This section describes the format of the individual records in the Abaqus results file.

The following record is written for each crack that is identified in the crack propagation analysis:

Record Format: Record Written for Crack Propagation Analysis

Record key	Record type	Attributes
1993(S)	Crack tip location and associated quantities	**1.** Crack number. **2.** Secondary surface (A8 format). **3.** Main surface (A8 format). **4.** Initial crack-tip node number. **5.** Current crack-tip node number. **6.** Flag to indicate crack propagation criterion. 1 for crack length criterion. 2 for critical stress criterion. 3 for crack opening displacement criterion. 5 for VCCT criterion. **7.** Cumulative incremental crack length. **8.** Value of σ_f if critical stress criterion is used. Current value of critical crack opening displacement if crack opening displacement criterion is used. **9.** Value of τ_f if critical stress criterion is used.

Records Written Once for Any File Output Request When Surfaces Are Defined in Abaqus/Standard

This section describes the format of the individual records in the Abaqus results file.

The number of data items for the following record depends on the type of surface being defined.

Rigid Surfaces

Record key	Record type	Attributes
1501(S)	Surface definition header	**1.** Surface name. **2.** Dimension key (1-1D, 2-2D, 3-3D, 4-Axisymmetric). **3.** Type key (1-Deformable, 2-Rigid). **4.** Number of facets making up the surface. **5.** Reference node label.

Deformable Surfaces

Record key	Record type	Attributes
1501(S)	Surface definition header	**1.** Surface name. **2.** Dimension key (1-1D, 2-2D, 3-3D, 4-Axisymmetric). **3.** Type key (1-Deformable, 2-Rigid). **4.** Number of facets making up the surface. **5.** Number of contact main surfaces associated with this surface through contact pairing (0 if this surface is a main surface). **6.** First main surface name. **7.** Second main surface name. **8.** Etc.
1502(S)	Surface facet	**1.** Underlying element number. **2.** Element face key (1–S1, 2–S2, 3–S3, 4–S4, 5–S5, 6–S6, 7–SPOS, 8–SNEG). **3.** Number of nodes in facet. **4.** Node number of the facet's first node. **5.** Node number of the facet's second node. **6.** Etc.

Records Written for Any Contact Surface File Output Request

This section describes the format of the individual records in the Abaqus results file.

Record Format: Records Written for Any Contact Surface File Output Request

Record key	Record type	Attributes
5(S)	Solution-dependent state variables Output variable: SDV	**1.** State variable 1. **2.** State variable 2. **3.** Etc. The record can have up to 80 words in ASCII format or 512 words in binary format. Repeat this record as often as necessary to output all active state variables in the model.
1503(S)	Output request definition	**1.** Contact file output (0). **2.** Secondary surface name. **3.** Main surface name. **4.** Node set containing a subset of the nodes making up the secondary surface.
1504(S)	Node header	**1.** Node number. **2.** Number of traction components (2 for 2D or axisymmetric cases, 3 for 3D cases).
1511(S)	Contact tractions Output variable: CSTRESS	**1.** Contact pressure between the node on the secondary surface and the main surface with which it interacts. **2.** Frictional shear traction component in the local 1-direction on the main surface. **3.** Frictional shear traction component in the local 2-direction on the main surface for 3D.
1512(S)	Viscous tractions Output variable: CDSTRESS	**1.** Viscous pressure between the node on the secondary surface and the main surface with which it interacts. **2.** Viscous shear traction component in the local 1-direction on the main surface. **3.** Viscous shear traction component in the local 2-direction on the main surface for 3D.
1521(S)	Contact clearances Output variable: CDISP	**1.** Separation of the surfaces in the direction of the normal to the main surface.

Record	Description	Attributes
		2. Accumulated relative tangential displacement of the surfaces in the local 1-direction on the main surface. **3.** Accumulated relative tangential displacement of the surfaces in the local 2-direction on the main surface for 3D.
1522$^{(S)}$	Total force due to contact pressure Output variable: CFN	**1.** Magnitude. **2.** Force component in the global 1-direction. **3.** Force component in the global 2-direction. **4.** Force component in the global 3-direction.
1523$^{(S)}$	Total force due to frictional stress Output variable: CFS	**1.** Magnitude. **2.** Force component in the global 1-direction. **3.** Force component in the global 2-direction. **4.** Force component in the global 3-direction.
1575$^{(S)}$	Total force due to contact pressure and frictional stress Output variable: CFT	**1.** Magnitude. **2.** Force component in the global 1-direction. **3.** Force component in the global 2-direction. **4.** Force component in the global 3-direction.
1524$^{(S)}$	Total area in contact Output variable: CAREA	**1.** Magnitude.
1526$^{(S)}$	Total moment about the origin due to contact pressure Output variable: CMN	**1.** Magnitude. **2.** Moment component about the global 1-axis. **3.** Moment component about the global 2-axis. **4.** Moment component about the global 3-axis.
1527$^{(S)}$	Total moment about the origin due to frictional stress Output variable: CMS	**1.** Magnitude. **2.** Moment component about the global 1-axis. **3.** Moment component about the global 2-axis. **4.** Moment component about the global 3-axis.
1576$^{(S)}$	Total moment about the origin due to contact pressure and frictional stress Output variable: CMT	**1.** Magnitude. **2.** Moment component about the global 1-axis. **3.** Moment component about the global 2-axis. **4.** Moment component about the global 3-axis.

1578(S)	Maximum torque that can be transmitted about the z-axis by a contact surface in an axisymmetric analysis with a friction coefficient of unity Output variable: CTRQ	**1.** Magnitude.
1573(S)	Coordinates of the center of the force due to contact pressure Output variable: XN	**1.** Coordinate in the global 1-direction. **2.** Coordinate in the global 2-direction. **3.** Coordinate in the global 3-direction.
1574(S)	Coordinates of the center of the force due to frictional stress Output variable: XS	**1.** Coordinate in the global 1-direction. **2.** Coordinate in the global 2-direction. **3.** Coordinate in the global 3-direction.
1577(S)	Coordinates of the center of the force due to contact pressure and frictional stress Output variable: XT	**1.** Coordinate in the global 1-direction. **2.** Coordinate in the global 2-direction. **3.** Coordinate in the global 3-direction.
1528(S)	Heat flux density Output variable: HFL	**1.** Magnitude.
1529(S)	HFL multiplied by the nodal area Output variable: HFLA	**1.** Magnitude.
1530(S)	Time integrated HFL Output variable: HTL	**1.** Magnitude.
1531(S)	Time integrated HFLA Output variable: HTLA	**1.** Magnitude.
1532(S)	Heat flux density due to frictional dissipation Output variable: SFDR	**1.** Magnitude.
1533(S)	SFDR multiplied by the nodal area Output variable: SFDRA	**1.** Magnitude.
1534(S)	Time integrated SFDR Output variable: SFDRT	**1.** Magnitude.
1535(S)	Time integrated SFDRA Output variable: SFDRTA	**1.** Magnitude.
1536(S)	Weighting factor Output variable: WEIGHT	**1.** Magnitude.
1537(S)	Heat flux density due to electrical current Output variable: SJD	**1.** Magnitude.
1538(S)	SJD multiplied by the nodal area Output variable: SJDA	**1.** Magnitude.

1539(S)	Time integrated SJD Output variable: SJDT	**1.** Magnitude.
1540(S)	Time integrated SJDA Output variable: SJDTA	**1.** Magnitude.
1541(S)	Electrical current density Output variable: ECD	**1.** Magnitude.
1542(S)	ECD multiplied by area Output variable: ECDA	**1.** Magnitude.
1543(S)	Time integrated ECD Output variable: ECDT	**1.** Magnitude.
1544(S)	Time integrated ECDA Output variable: ECDTA	**1.** Magnitude.
1545(S)	Pore fluid volume flux per unit area Output variable: PFL	**1.** Magnitude.
1546(S)	PFL multiplied by the nodal area Output variable: PFLA	**1.** Magnitude.
1547(S)	Time integrated PFL Output variable: PTL	**1.** Magnitude.
1548(S)	Time integrated PFLA Output variable: PTLA	**1.** Magnitude.
1549(S)	Total pore fluid volume flux leaving the secondary surface Output variable: TPFL	**1.** Magnitude.
1550(S)	Time integrated TPFL Output variable: TPTL	**1.** Magnitude.

Records for Bond Failure Quantities from Crack Propagation Analysis

Record key	**Record type**	**Attributes**
1570(S)	Time when bond failure occurs Output variable: DBT	**1.** Magnitude.
1571(S)	Fraction of stress that remains at bond failure Output variable: DBSF	**1.** Magnitude.
1572(S)	Remaining stress in the failed bond Output variable: DBS	**1.** 11-component of debond stress. **2.** 12-component of debond stress.

290$^{(S)}$	Relative displacement behind crack when fracture criterion is met Output variable: OPENBC	**1.** Magnitude.
293$^{(S)}$	Effective energy release rate ratio Output variable: EFENRRTR	**1.** Magnitude.
294$^{(S)}$	Bond state (varies from 1.0 to 0.0) Output variable: BDSTAT	**1.** Magnitude.
235$^{(S)}$	Damage variable Output variable: CSDMG	**1.** Magnitude.
295$^{(S)}$	Critical stress at failure Output variable: CRSTS	**1.** 11-component of critical stress. **2.** 12-component of critical stress. **3.** 13-component of critical stress (only available to three-dimensional models).
296$^{(S)}$	Strain energy release rate Output variable: ENRRT	**1.** 11-component of strain energy release rate. **2.** 12-component of strain energy release rate. **3.** 13-component of strain energy release rate (only available to three-dimensional models).

Record for Surface-Based Pressure Penetration Analysis

Record key	Record type	Attributes
1592$^{(S)}$	Fluid pressure for surface-based pressure penetration analysis Output variable: PPRESS	**1.** Magnitude.

Records for Surface-Based Cohesive Behavior with Damage

Record key	Record type	Attributes
253$^{(S)}$	Overall value of the scalar damage variable Output variable: CSDMG	**1.** Magnitude.
345$^{(S)}$	Maximum contact stress damage initiation criterion Output variable: CSMAXSCRT	**1.** Magnitude.
346$^{(S)}$	Maximum separation damage initiation criterion Output variable: CSMAXUCRT	**1.** Magnitude.

347$^{(S)}$	Quadratic contact stress damage initiation criterion Output variable: CSQUADSCRT	**1.** Magnitude.
348$^{(S)}$	Quadratic separation damage initiation criterion Output variable: CSQUADUCRT	**1.** Magnitude.

Records Written Once for Any File Output Request When Cavities Are Defined

This section describes the format of the individual records in the Abaqus results file.

Record Format: Records Written Once for Any File Output Request When Cavities Are Defined

Record key	Record type	Attributes
1601$^{(S)}$	Cavity definition header	**1.** Number of surfaces making up the cavity. **2.** Cavity name. **3.** Name of cavity's first surface. **4.** Name of cavity's second surface. **5.** Etc.
1610$^{(S)}$	Facet order record size	**1.** Maximum record length (including the record length and record key words) for cavity facet order records that follow. The cavity facet order data will be subdivided into multiple records as needed to fit within this maximum length. The record key for any continuation record will be the same as for the first record.
1602$^{(S)}$	Cavity facet order	**1.** Number of facets making up the cavity. **2.** Cavity name. **3.** Cavity's first (underlying) element number. **4.** First element face key (1-S1, 2–S2, 3–S3, 4–S4, 5–S5, 6–S6, 7–SPOS, 8–SNEG) **5.** Cavity's second (underlying) element number. **6.** Second element face key (1–S1, 2–S2, 3–S3, 4–S4, 5–S5, 6–S6, 7–SPOS, 8–SNEG) **7.** Etc.

Records Written for Any View Factor Matrix Output Request

This section describes the format of the individual records in the Abaqus results file.

The ordering of the facets (each facet corresponds to one row of the view factor matrix) is that appearing in the cavity facet order record 1602.

Record Format: Records Written for Any View Factor Matrix Output Request

Record key	Record type	Attributes
$1608^{(S)}$	Output request definition	**1.** View factor output (0). **2.** Cavity name.
$1605^{(S)}$	View factor matrix header	**1.** Number of facets in the cavity. **2.** Cavity name.
$1609^{(S)}$	View factor matrix record size	**1.** Maximum record length (including the record length and record key words) for view factor matrix and facet area records that follow. The matrix or facet area records will be subdivided into multiple records as needed to fit within this maximum length. The record key for any continuation record will be the same as for the first record.
$1606^{(S)}$	Nonsymmetric view factor matrix	**1.** (1, 1) dimensionless view factor. **2.** (1, 2) dimensionless view factor. **3.** (1, 3) dimensionless view factor. **4.** Etc., stored in rows.
$1607^{(S)}$	Facet areas	**1.** Area of first facet. **2.** Area of second facet. **3.** Area of third facet. **4.** Etc.

Records Written for Any Radiation File Output Request

This section describes the format of the individual records in the Abaqus results file.

Record Format: Records Written for Any Radiation File Output Request

Record key	Record type	Attributes
1603(S)	Output request definition	**1.** Radiation file output (1). **2.** Cavity name. **3.** Surface name. **4.** Element set name.
1604(S)	Facet header record	**1.** (Underlying) user element number. **2.** Element face key (1–S1, 2–S2, 3–S3, 4–S4, 5–S5, 6–S6, 7–SPOS, 8–SNEG) **3.** Facet area.
231(S)	Radiation flux density	**1.** Magnitude.
232(S)	Radiation flux	**1.** Magnitude.
233(S)	Time integrated radiation flux density	**1.** Magnitude.
234(S)	Time integrated radiation flux	**1.** Magnitude.
235(S)	Total view factor (sum of view factor matrix row)	**1.** Magnitude.
236(S)	Facet temperature	**1.** Magnitude.

Records Written for Any Section File Output Request

This section describes the format of the individual records in the Abaqus results file.

The output variables described below are not available for random response analysis.

Record Format: Records Written for Any Section File Output Request

Record key	Record type	Attributes
1580$^{(S)}$	Output request definition	**1.** Surface section output (1). **2.** Section name.
1581$^{(S)}$	Section output header record	**1.** Surface name. **2.** System of coordinates used for output (1–Global, 2–Local). **3.** Flag to indicate whether or not the local coordinate system and the output are updated during the analysis (1–Yes, 2–No).

For All Analysis Types

Record key	Record type	Attributes
1582$^{(S)}$	Global coordinates of the anchor point	**1.** First coordinate. **2.** Etc.
1583$^{(S)}$	Direction cosines of the local coordinate system	**1.** First component of the first direction. **2.** Second component of the first direction. **3.** Third component of the first direction. **4.** First component of the second direction. **5.** Second component of the second direction. **6.** Third component of the second direction.
1584$^{(S)}$	Area of the defined section Output variable: SOAREA	**1.** Magnitude.

For Stress/Displacement Analyses

Record key	Record type	Attributes
1585$^{(S)}$	Total force in the section in the selected system	**1.** Magnitude. **2.** First force component.

Record key	Record type	Attributes
	Output variable: SOF	**3.** Etc.
1586(S)	Total moment in the section about the origin of the selected system Output variable: SOM	**1.** Magnitude. **2.** First moment component. **3.** Etc.
1587(S)	Global coordinates of the center of the total force in the section Output variable: SOCF	**1.** First coordinate. **2.** Etc.

For Heat Transfer Analyses

Record key	Record type	Attributes
1588(S)	Total heat flux across the section Output variable: SOH	**1.** Magnitude.

For Electrical Analyses

Record key	Record type	Attributes
1589(S)	Total current across the section Output variable: SOE	**1.** Magnitude.

For Mass Diffusion Analyses

Record key	Record type	Attributes
1590(S)	Total mass flow across the section Output variable: SOD	**1.** Magnitude.

For Coupled Pore Fluid Diffusion-Stress Analyses

Record key	Record type	Attributes
1591(S)	Total pore fluid volume flux across the section Output variable: SOP	**1.** Magnitude.

Procedure type keys

Overview

The table below provides a description of the procedure type keys.

Table 1: Keys to procedure types.

Key	Description
1	Static, automatic incrementation
2	Static, direct incrementation
4	Direct cyclic, automatic time incrementation
5	Direct cyclic, fixed time incrementation
11	Implicit dynamic, half-increment residual tolerance given
12	Implicit dynamic, fixed time increments
13	Implicit dynamic, subspace projection
17	Explicit dynamic
21	Quasi-static, explicit time integration
22	Quasi-static, implicit integration
31	Heat transfer, steady-state
32	Heat transfer, transient, fixed time increments
33	Heat transfer, transient, maximum allowable nodal temperature change given
34	Mass diffusion, steady-state
35	Mass diffusion, transient, fixed time increments
36	Mass diffusion, transient, maximum allowable normalized concentration change given
41	Eigenvalue frequency extraction
42	Eigenvalue buckling prediction
51	Substructure generation
61	Geostatic stress field
62	Coupled pore fluid diffusion/stress, steady-state, fixed time incrementation
63	Coupled pore fluid diffusion/stress, steady-state, automatic time incrementation
64	Coupled pore fluid diffusion/stress, transient, fixed time incrementation
65	Coupled pore fluid diffusion/stress, transient, automatic time incrementation
71	Coupled thermal-stress, steady-state
72	Coupled thermal-stress, transient, fixed time increments
73	Coupled thermal-stress, transient, maximum allowable nodal temperature change and/or accuracy tolerance parameter given
74	Explicit dynamic coupled thermal-stress
75	Coupled thermal-electrical, steady-state
76	Coupled thermal-electrical, transient analysis, fixed time increments
77	Coupled thermal-electrical, transient analysis, maximum allowable nodal temperature change given
85	Steady-state transport, automatic incrementation
86	Steady-state transport, direct incrementation
91	Response spectrum

Key	Description
92	Modal dynamic
93	Steady-state dynamic
94	Random response
95	Direct-solution steady-state dynamic
98	Annealing
101	Time harmonic electromagnetic
102	Coupled electrical-temperature-displacement, steady-state
103	Coupled electrical-temperature-displacement, transient, fixed time increments
104	Coupled electrical-temperature-displacement, transient, automatic incrementation

Accessing the Results File Information

Products: Abaqus/Standard Abaqus/Explicit

References:

- *About the Results File*
- *Results File*
- *Utility Routines for Accessing the Results File*
- *Postprocessing of Abaqus Results*

Overview

The Abaqus results (`.fil`) file is written using internal data management routines to minimize I/O cost. A postprocessing program must use these same Abaqus data management routines to read the results file. The following utility routines must be called to obtain data from the Abaqus results file:

- `INITPF`
- `DBRNU`
- `DBFILE`
- `POSFIL`

You can also write a file in the format of the Abaqus results file by using the following utility subroutines:

- `INITPF`
- `DBFILW`

The syntax of these utility subroutines is described in *Utility Routines for Accessing the Results File*.

Reading Floating Point and Integer Variables

To read both floating point and integer variables in the records, the following coding can be used in the postprocessing program:

```
INCLUDE 'aba_param.inc'
      DIMENSION ARRAY(513), JRRAY(NPRECD,513)
      EQUIVALENCE (ARRAY(1),JRRAY(1,1))
```

With this technique, for example, the record key is available after each call to `DBFILE` with `LOP`=0 as

```
KEY =  JRRAY (1,2)
```

The use of `aba_param.inc` eliminates the need to have different versions of the code for single and double precision. The file `aba_param.inc` defines an appropriate `IMPLICIT REAL` statement and sets the value of `NPRECD` to 1 or 2, depending upon whether the machine uses single or double precision. The file `aba_param.inc` is referenced from the `site` subdirectory of the Abaqus installation when the postprocessing program is compiled and linked using the **abaqus make** utility (explained below).

Linking the Postprocessing Program

The postprocessing program must be linked using the **make** parameter when running the Abaqus execution procedure (see *Making User-Defined Executables and Subroutines*). To link properly, the postprocessing program cannot contain a Fortran PROGRAM statement. Instead, the program must begin with a Fortran SUBROUTINE with the name ABQMAIN.

Compiling, linking, and running a postprocessing program consists of two steps. For example, if the name of the postprocessing program is `postproc.f`, use the following command to compile and link `postproc.f`:

```
abaqus make job=postproc
```

The program must then be run using the command:

```
abaqus postproc
```

Calling the Utility Subroutines for Reading the Results File

Subroutine `INITPF` must be called before any results file is accessed. This subroutine contains Fortran OPEN statements for all Fortran units assigned to results files through the call to `INITPF`; therefore, your code must not contain any OPEN statements for these units. Abaqus constructs a file name for a given unit based on information supplied as `LRUNIT(1,K1)` and `FNAME`, as discussed in *Utility Routines for Accessing the Results File*.

Subroutine `DBRNU` must also be called before reading the first results file and then again each time you need to change to reading another results file. This subroutine simply establishes the Fortran unit number of the results file being read; no information is returned. `DBRNU` can be called before or after `INITPF` but must be called before `DBFILE`.

Subroutine `DBFILE` is used to read each record from the results file. This subroutine will return one record at a time in the format described in *Results File*.

Example

The following program reads all the von Mises stresses in the results file and obtains the maximum value. Then, it prints this value along with the element, section point, and integration point numbers where it occurred.

In this example Fortran unit 8 is used to read the results file, and the name of the results file is assumed to be `TEST.fil`. The results file is assumed to be a binary file, and only one results file will be read. Thus, `LRUNIT` is dimensioned as `LRUNIT(2,1)`; and in the call to the `INITPF` routine `NRU` is set to 1, `LRUNIT(1,1)` is set to 8, and `LRUNIT(2,1)` is set to 2. A new results file will not be written, so `LOUTF` is set to zero.

```
      SUBROUTINE ABQMAIN
C     Calculate the maximum von Mises stress and its location
C
      INCLUDE 'aba_param.inc'
      CHARACTER*80 FNAME
      DIMENSION ARRAY(513),JRRAY(NPRECD,513),LRUNIT(2,1)
      EQUIVALENCE (ARRAY(1),JRRAY(1,1))
C
C     File initialization
C
      FNAME='TEST'
      NRU=1
      LRUNIT(1,1)=8
      LRUNIT(2,1)=2
      LOUTF=0
      CALL INITPF(FNAME,NRU,LRUNIT,LOUTF)
```

```
      JUNIT=8
      CALL DBRNU(JUNIT)
C
C     Loop on all records in results file
C
      STRESS=0.
      DO 100 K1=1,99999
C
        CALL DBFILE(0,ARRAY,JRCD)
        IF(JRCD.NE.0)GO TO 110
        KEY=JRRAY(1,2)
C
        IF(KEY.EQ.1) THEN
C
C          Element header record:
C          extract element, sec pt, int pt numbers
C
           JEL=JRRAY(1,3)
           JPNT=JRRAY(1,4)
           JSPNT=JRRAY(1,5)
C
C     Stress invariant record for Abaqus/Standard
        ELSE IF(KEY.EQ.12)THEN
C     Stress invariant record for Abaqus/Explicit
        ELSE IF(KEY.EQ.75)THEN
C
C          Extract von Mises stress
C
           IF(ARRAY(3).GT.STRESS)THEN
              STRESS=ARRAY(3)
              KEL=JEL
              KPNT=JPNT
              KSPNT=JSPNT
           END IF
        END IF
C
 100  CONTINUE
 110  CONTINUE
C
      WRITE(6,120) KEL,KPNT,KSPNT,STRESS
 120  FORMAT(5X,'ELEMENT',I5,5X,'POINT',I4,5X,'SECTION POINT',
     1 I4,5X,'STRESS',1PG12.3)
      STOP
      END
```

See *Postprocessing of Abaqus Results* for additional examples.

Writing a File in the Results File Format

Subroutine `DBFILW` can be used to write a file in the format of the Abaqus results file to modify the file information or to add additional information before postprocessing. Subroutine `INITPF` must be called before `DBFILW`.

The file will be written to Fortran unit 9 with the extension `.fin`. Unit 9 is opened by Abaqus when `DBFILW` is first called; your coding must not open or redefine unit 9, but you must ensure that Fortran unit 9 is saved following the job.

Joining data from multiple results files and converting file format: FJOIN contains an example of the use of subroutine `DBFILW` to merge specific records of discontinuous results files. Continuous results files are required for postprocessing purposes; if you have written a results file during an analysis and a new results file on the restart of the analysis without making the files continuous, they must be made continuous before postprocessing. *Analysis of a cantilever subject to earthquake motion* also shows the use of `DBFILW` for merging results files. Alternatively, results files can be merged using the **abaqus append** utility as described in *Joining Results (.fil) Files*.

The `DBFILW` subroutine can also be used to convert the Abaqus results file from binary to ASCII format to transfer it from one computer system to another. Alternatively, this conversion can be done automatically by using the **abaqus ascfil** execution procedure, as described in *ASCII Translation of Results (.fil) Files*.

Utility Routines for Accessing the Results File

Products: Abaqus/Standard Abaqus/Explicit

References:

- *Accessing the Results File Information*
- *URDFIL*
- *Joining data from multiple results files and converting file format: FJOIN*
- *Calculation of principal stresses and strains and their directions: FPRIN*
- *Creation of a perturbed mesh from original coordinate data and eigenvectors: FPERT*

Overview

The Abaqus results (`.fil`) file can be accessed with the utility routines described in this section.
Access is subsequent to an analysis by a user-written postprocessing program or, in Abaqus/Standard, from within an analysis by user subroutine *URDFIL*.

Only the subroutines `DBFILE` and `POSFIL` can be called from user subroutine *URDFIL*.

`DBFILE` (Read from a File)

Utility Routine Interface

```
CALL DBFILE(LOP,ARRAY,JRCD)
```

Variables to Be Provided to the Utility Routine

LOP

A flag, which you must set before calling `DBFILE`, indicating the operation. Set `LOP`=0 to read the next record in the file; set `LOP`=2 to rewind the file currently being read (for example, if it is necessary to read the file more than once, it must be rewound since it is a sequential file). If `LOP`=2 is used, the file must first be read to the end, and it should be rewound only when the end-of-file is reached.

Variables Returned from the Utility Routine

ARRAY

The array containing one record from the file, in the format described in *Results File*. When `LOP`=0, this array will be filled by the data management routines with the contents of the next record in the file as each call to `DBFILE` is executed. `ARRAY` must be dimensioned adequately in your routines to contain the largest record in the file. For almost all cases 500 words is sufficient. The exceptions arise if the problem definition includes user elements or user materials that use more than this many state variables or if substructures with a large number of retained degrees of freedom are used (see *Using Substructures* for more details regarding substructures). When the results file has been written on a system on which Abaqus runs in double precision, `ARRAY` must be declared double precision in your routine.

JRCD

Returned as nonzero if an end-of-file marker is read when `DBFILE` is called with `LOP`=0.

`DBFILW` (Write to a File)

Utility Routine Interface

```
CALL DBFILW(LOP,ARRAY,JRCD)
```

Variables to Be Provided to the Utility Routine

ARRAY
The array containing one record to be written to the file, in the format described in *Results File*.

JRCD
Return code (0 – record written successfully, 1 – record not written).

LOP
Not currently used.

`DBRNU` (Set a Unit Number for a File)

Utility Routine Interface

```
CALL DBRNU(JUNIT)
```

Variables to Be Provided to the Utility Routine

JUNIT
The Fortran unit number of the results file to be read. Valid unit numbers are 8 to read the `.fil` file, 15–18, or numbers greater than 100.

`INITPF` (Initialize a File)

Utility Routine Interface

```
CALL INITPF(FNAME,NRU,LRUNIT,LOUTF)
```

Variables to Be Provided to the Utility Routine

FNAME
A character string defining the root file name (that is, the name without an extension) of the files being read or written. `FNAME` must be declared as `CHARACTER*80` and can include the directory specification as well as the root file name. The extension of each individual file is defined by the `LRUNIT` array below. See the discussion below for file naming conventions.

NRU
An integer giving the number of results files that the postprocessing program will read. Normally only one results file is read, but sometimes it is necessary to read several results files—for example, to merge them into a single file.

LRUNIT
An integer array that must be dimensioned `LRUNIT(2,NRU)` in the postprocessing program and must contain the following data before `INITPF` is called:

LRUNIT(1,K1) is the Fortran unit number on which the K1th results file will be read. Valid unit numbers are 8 to read the .fil file, 15–18, or numbers greater than 100. All other units are reserved by Abaqus. See below for naming conventions based on the unit numbers.

LRUNIT(2,K1) is an integer that must be set to 2 if the K1th results file was written as a binary file or set to 1 if the K1th results file was written in ASCII format.

LOUTF
Needs to be defined only if the program that is making the call to INITPF will also write an output file in the Abaqus results file format (for example, if results files are being merged into a single results file or if a results file is being converted from binary to ASCII format). In that case LOUTF should be set to 2 if the output file is to be written as a binary file or set to 1 if the output file is to be written as an ASCII file. This results file will be written with the file name extension .fin. See *Accessing the Results File Information* for a discussion of writing results files; see below for information on the naming of this file.

File Naming Conventions

The file extension is derived from the value of LRUNIT(1,K1). If LRUNIT(1,K1) is 8, the file name will be constructed with the extension fil. Any other unit number will result in a file extension of 0nn, where nn is the number assigned to LRUNIT(1,K1). For example, if LRUNIT(1,K1) is 15, the file extension is .015. If an output file has been indicated by a nonzero value of LOUTF, its extension will be .fin.

For example, to read a file xxxx.fil, set LRUNIT(1,K1) to 8 and the character variable FNAME to xxxx using assignment or data statements. If desired, FNAME can include a directory specification, device name, or path. Operating system environment and shell variables will not be translated properly and, therefore, should not be used.

All error messages generated by Abaqus are written to Fortran unit 6. On most machines error messages will be printed by default directly to the screen if the program is run interactively. You can include an open statement for unit 6 in the main program to redirect messages to a file. If you wish to read or write to units other than those units specified in LRUNIT, OPEN statements for those units may have to be included in the program (depending upon the computer being used). Unit numbers of such auxiliary files should be greater than 100 to avoid any conflict with Abaqus internal files.

POSFIL (Determine Position in a File)

The POSFIL utility routine is available only in Abaqus/Standard.

Utility Routine Interface

```
CALL POSFIL(NSTEP,NINC,ARRAY,JRCD)
```

Variables to Be Provided to the Utility Routine

NSTEP
Desired step. If this variable is set to 0, the first available step will be read.

NINC
Desired increment. If this variable is set to 0, the first available increment of the specified step will be read.

Variables Returned from the Utility Routine

ARRAY

Real array containing the values of record 2000 from the results file for the requested step and increment.

JRCD

Return code (0 – specified increment found, 1 – specified increment not found). If the step and increment requested are not found in the results file, `POSFIL` will return an error and leave you positioned at the end of the results file.

Positioning with `POSFIL`

You may find it convenient to call `POSFIL` with both `NSTEP` and `NINC` set to 0 to skip over the information that is written to the results file at the beginning of an analysis (see *Results File*) and, thus, start reading from the first increment written to the file.

`POSFIL` cannot be used to move backward in the results file: you cannot use `POSFIL` to find a given increment in the file and then make a second call to `POSFIL` later to read an increment earlier than the first one found. If this is attempted, `POSFIL` will return an error indicating that the requested increment was not found.

Output Variable Indexes

These indexes list the output variables that apply to a particular Abaqus product.

In this section:

Abaqus/Standard Output Variable Index

This index lists the output variables available in Abaqus/Standard.

A	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*	*K*	*L*	*M*
N	*O*	*P*	*Q*	*R*	*S*	*T*	*U*	*V*	*W*	*X*	*Y*	Z

A

D

F

G

H

I

S

T

U

V

Abaqus/Explicit Output Variable Index

This index lists the output variables available in Abaqus/Explicit.

A

E

J

L

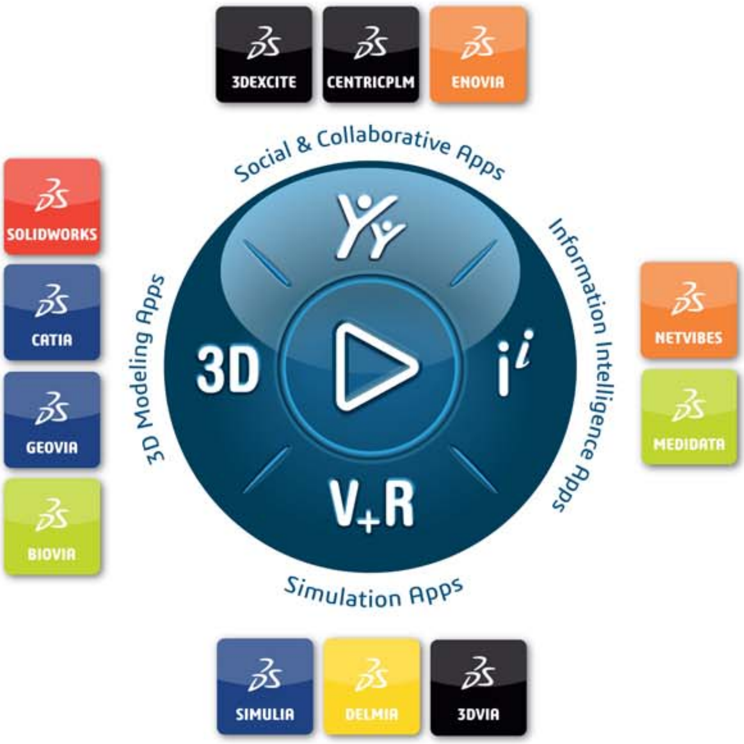
M

N

Q

R

S



Our **3D**EXPERIENCE® platform powers our brand applications, serving 12 industries, and provides a rich portfolio of industry solution experiences.

Dassault Systèmes is a catalyst for human progress. We provide business and people with collaborative virtual environments to imagine sustainable innovations. By creating virtual twin experiences of the real world with our **3D**EXPERIENCE platform and applications, our customers can redefine the creation, production and life-cycle-management processes of their offer and thus have a meaningful impact to make the world more sustainable. The beauty of the Experience Economy is that it is a human-centered economy for the benefit of all – consumers, patients and citizens.

Dassault Systèmes brings value to more than 300,000 customers of all sizes, in all industries, in more than 150 countries. For more information, visit **www.3ds.com**.

Europe/Middle East/Africa
Dassault Systèmes
10, rue Marcel Dassault
CS 40501
78946 Vélizy-Villacoublay Cedex
France

Asia-Pacific
Dassault Systèmes
17F, Foxconn Building,
No. 1366, Lujiazui Ring Road
Pilot Free Trade Zone, Shanghai 200120
China

Americas
Dassault Systèmes
175 Wyman Street
Waltham, Massachusetts
02451-1223
USA

DASSAULT SYSTEMES