

SAND92-2292  
Unlimited Release  
Printed January 1993  
Reprinted October 1993  
Reprinted August 1994

Distribution  
Category UC-705

# Overview of the Sandia National Laboratories Engineering Analysis Code Access System

Gregory D. Sjaardema  
Solid and Structural Mechanics Department  
Sandia National Laboratories  
Albuquerque, NM 87185

## Abstract

The Sandia National Laboratories (SNL) Engineering Analysis Code Access System (SEACAS) is a collection of structural and thermal codes and utilities used by analysts at SNL. The system includes pre- and post-processing codes, analysis codes, database translation codes, support libraries, UNIX\* shell scripts, and an installation system.

SEACAS is used at SNL on a daily basis as a production, research, and development system for the engineering analysts and code developers. Approximately 180 days of Cray Y-MP† CPU time per year are used at SNL by SEACAS codes. The job mix includes jobs using only a few seconds of CPU time, up to jobs using ten days of CPU time. SEACAS is running on several different systems at SNL including Cray Unicos, Hewlett Packard HP-UX‡, Digital Equipment Ultrix§, Sun SunOS\*\*, IBM AIX††, and Silicon Graphics IRIX‡‡.

This document is a short description of the codes in the SEACAS system.

---

\*UNIX is a registered trademark of UNIX Systems Laboratories Inc.

†Cray and Unicos are registered trademarks of Cray Research, Inc.

‡HP-UX is a registered trademark of Hewlett-Packard Company.

§Ultrix is a registered trademark of Digital Equipment Corporation.

\*\*Sun and SunOS are registered trademarks of Sun Microsystems, Inc.

††IBM and AIX are registered trademarks of International Business Machines Corporation.

‡‡IRIX is a registered trademark of Silicon Graphics, Inc.

## *Table of Contents*

---

<b>Philosophy</b> .....	<b>3</b>
Modular Structure of SEACAS .....	3
Expanded Modular View of SEACAS .....	3
Structure of Mesh Generation Process .....	4
SEACAS Code Types .....	4
Preprocessing and Postprocessing Codes .....	4
Analysis Programs .....	5
Translation Codes .....	5
Support Libraries .....	6
Scripts .....	6
Constitutive Models .....	6
<b>SEACAS Abstracts and References</b> .....	<b>7</b>
Algebra .....	7
Aprepro .....	7
Blot .....	7
Conchas .....	8
Cubit .....	8
Exodus I and Genesis File Format .....	8
Exodus II File Format .....	8
Fastq .....	9
Gen3D .....	9
Genshell .....	10
Gjoin .....	10
Grepos .....	10
Grove .....	11
Jac2D, Jac3D, and Jacq3D .....	11
Merlin II .....	12
Numbers .....	12
Pronto2D and Pronto3D .....	12
Sancho .....	14
Santos and Santos3D .....	14
Supes Library .....	15
Constitutive Models .....	15
<b>References</b> .....	<b>16</b>
<b>Modifications</b> .....	<b>16</b>

## *List of Figures*

---

Modular Structure of SEACAS .....	4
Expanded Modular View of SEACAS .....	4
Structure of Mesh Generation Process.....	5

# Philosophy

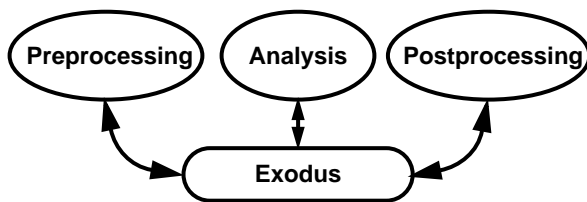
SEACAS is a modular system based upon a common binary datafile format called EXODUS<sup>1</sup> which includes the mesh description and the timeplanes of the computed results.

A subset of this format, called GENESIS<sup>2</sup>, was a precursor to the EXODUS format. GENESIS is often used in references and reports to refer to the mesh description portion of the EXODUS format.

We are currently upgrading the original EXODUS file format. EXODUS now refers to a generic format. When it is important to differentiate, the older sequential database is called EXODUS I and the newer randomly accessible, platform independent database is known as EXODUS II<sup>3</sup>.

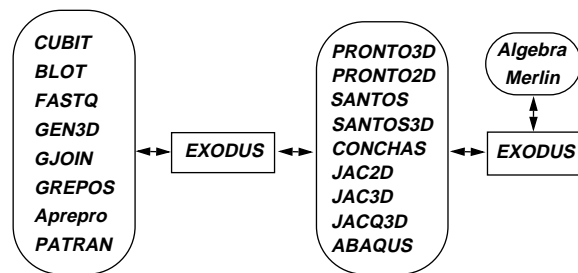
Some codes have not yet migrated to the EXODUS II file format. The scripts that control the codes will automatically translate the input files to the required format. so that all of the preprocessing, analysis, and postprocessing codes can read and/or write the EXODUS database files formats as required. A schematic of this is shown in Figure 1.

FIGURE 1. Modular Structure of SEACAS



With this structure, codes can be tailored for a single function. For example, an analysis code can be added to the system without writing new mesh generation and post processing programs. Also, code to code data transfers and restarts use the EXODUS format. A more complete view of this is shown in Figure 2.

FIGURE 2. Expanded Modular View of SEACAS

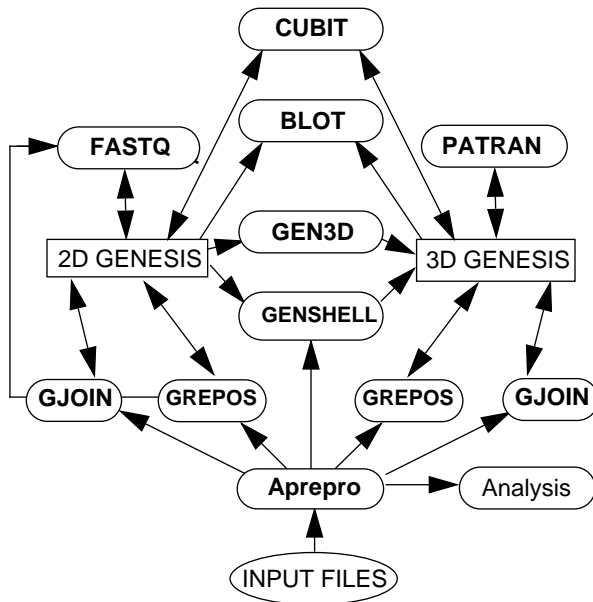


This modular concept is used extensively in the mesh generation process. Several special-purpose codes have been written that each perform a certain mesh generation task. Examples of this are:

- generate two-dimensional mesh (fastq).
- transform two-dimensional mesh into a three-dimensional mesh (gen3d, genshell).
- join two or more 2D or 3D meshes into a single mesh (gjoin).
- reposition a 2D or 3D mesh (grepos).

This structure is shown in Figure 3 on the following page. Finite element meshes of complicated three-dimensional geometries have been successfully generated using this system built up of codes that singly provide a limited capability, but when used as a system are very powerful.

FIGURE 3. Structure of Mesh Generation Process



All of the codes are written in as portable a form as possible. Fortran codes are written in ANSI Standard FORTRAN-77<sup>4</sup> and C language codes are written in ANSI Standard C<sup>5</sup> where possible. Machine-specific routines are limited in number and are grouped together to minimize the time required to adapt them to a new system. SEACAS codes have been ported to several Unix systems including Cray, HP, Sun, Dec, Stardent, Convex, IBM, and SGI.

A code management system called `cvs`<sup>6</sup> is used for all of the SEACAS codes to provide traceability and retrievability for Quality Assurance. The change logs include who changed the code, when it was changed, what was changed, and why the change was made. If required, any previous version can be retrieved.

## SEACAS Code Types

SEACAS is divided into five broad categories of codes: prepost, analysis, translators, libraries, and scripts. The categories can be roughly defined as follows:

**prepost:** Pre- and Postprocessing codes including mesh generation, visualization, preprocessors, and database manipulation codes.

**analysis:** Finite element analysis codes including quasistatic, transient dynamics, thermal, and electro-mechanics. Two- and three-dimensional, nonlinear, large deformation.

**translators:** Translation codes for editing and viewing output files, inter-machine translation, EXODUS I from/to EXODUS II translation, and EXODUS I from/to commercial database translation.

**libraries:** Support libraries including database routines, common machine-specific routines, plot routines, graphics device drivers, and interactive help routines.

**scripts:** Unix shell scripts for executing the prepost, analysis, and translation codes. Also includes support and installation routines.

## Preprocessing and Postprocessing Codes

The pre- and postprocessing codes are comprised of the following mesh generation, visualization, and database manipulation codes.

**aprepro:** An algebraic preprocessing program.

**blot:** The primary graphical two-dimensional and three-dimensional postprocessing code. It includes deformed mesh plots, contour plots, shaded fringe plots, variable versus variable and time history plots, and distance versus variable plots.

**cubit:** Interactive finite element mesh generation tool for creating quadrilateral and hexahedral elements. Contains several two- and three mesh generation meshing algorithms, including mapping and paving, for automated mesh generation.

**fastq:** Interactive, unstructured two-dimensional quadrilateral mesh generation program. Includes the paving algorithm for general surface filling with arbitrary sizing, as well as other meshing options.

**gen3d:** Transform a two-dimensional database into a three-dimensional GENESIS database. Several transformations are supported and additional transformations can be easily added.

**genshell:** Transform a two-dimensional GENESIS database into a three-dimensional shell GENESIS database. Several transformations are supported and additional transformations can be easily added.

**gjoin:** Join together two or more GENESIS databases into a single GENESIS database.

**grepos:** Transforms the geometry of a GENESIS database by scaling, mirroring, offsetting, or rotating. It can also modify the database by deleting or renaming material blocks, sideset identifications, or nodeset identifications.

**grope:** Interactively examine an EXODUS database. Grope is also contained within the blot program. Grope is primarily used to validate EXODUS files.

**numbers:** Calculates several properties of an EXODUS file, including mass properties, timesteps, condition numbers, cavity volumes, and others.

## Analysis Programs

The analysis codes include quasistatic, transient dynamics, thermal, and electromechanics codes.

**conchas:** Conchas is a linear structural analysis code for axisymmetric structures with loads that are symmetric about a plane.

**jac2d:** Jac2d is a finite element program which uses a nonlinear conjugate gradient technique to solve the large displacement, large strain, temperature dependent and material nonlinear problem for two-dimensional plane or axisymmetric solids.

**jac3d:** Jac3d is a finite element program which uses a nonlinear conjugate gradient technique to solve the large displacement, large strain, temperature dependent and material nonlinear problem for three-dimensional solids.

**jacq3d:** Jacq3d is a finite element program which uses a nonlinear conjugate gradient technique to solve the thermal conduction problem for three-dimensional solids.

**pronto2d:** Pronto2d is a two-dimensional (planar and axisymmetric) transient solid dynamics code. Lagrangian formulation with explicit time integra-

tion is used for analyzing large deformations of highly nonlinear materials subjected to extremely high strain rates.

**pronto3d:** Pronto3d is a three-dimensional transient solid dynamics code. Lagrangian formulation with explicit time integration is used for analyzing large deformations of highly nonlinear materials subjected to extremely high strain rates.

**sancho:** Sancho is a finite element program which uses a dynamic relaxation technique to compute the quasistatic, large deformation, temperature dependent, inelastic response of planar or axisymmetric solids.

**santos:** Santos is a two-dimensional finite element program which uses a dynamic relaxation technique to compute the quasistatic, large deformation, temperature dependent, inelastic response of planar or axisymmetric solids. Newer than sancho.

**santos3d:** Santos3d is a three-dimensional version of santos. It is currently in development.

## Translation Codes

The translation codes are used to convert EXODUS databases into different formats and to edit EXODUS databases.

**abaexo:** Converts ABAQUS (commercial finite element code) results files to EXODUS I.

**algebra:** Manipulates EXODUS finite element output data by evaluating algebraic expressions. Equation variables are dependent on the input database variable names.

**conex:** Concatenates several EXODUS I files into a single EXODUS I file. Used to create a single EXODUS file from analyses that have been restarted.

**exoexo:** A programs which simply translate an EXODUS I file to the same EXODUS I file. It is used as a base program for writing new translators or database manipulation programs.

**ex1ex2v2** and **ex2ex1v2:** Converts an EXODUS file to its alternate format. **exo1exo2** and **exo2exo1** are other format translators with special application.

**exogen:** Creates a GENESIS mesh database from a specific time step of an EXODUS I file. Used when an analysis of a deformed geometry is required. For example, an impact analysis followed by a thermal analysis.

**exopat** and **patexo**: Converts EXODUS I to and from PATRAN (commercial mesh generation program) neutral file format.

**exotxt** and **txtexo**: Converts EXODUS I to and from a specially formatted ASCII text file.

**exoxdr** and **xdrexo**: Converts EXODUS I to and from an external data representation format (XDR) which can be transmitted between computers of different architectures, word lengths, and byte orders.

**merlin2**: Transfers (maps) nodal data between finite element meshes. For example, thermal output from a heat conduction code to a thermal input file for a quasistatic mechanics code.

**seaexo**: Converts SEACO files (the binary file format that preceded EXODUS) into EXODUS I files.

In addition to the translation codes listed above, the Cray Unicos system provides the capability to translate EXODUS I files into a VMS and IEEE format using the **exoexo** code. EXODUS II files, being machine independent, need no such translations.

### Support Libraries

The support libraries include routines which are used in several of the codes.

**exodusii**: A library of application interface routines used to store and retrieve EXODUS II files.

**netcdf**: Public domain database routines used to handle low-level data storage for EXODUS II files. Copyrighted, netCDF ©1993 University Corporation for Atmospheric Research/Unidata.

**supes**: A library of frequently used non-numerical routines for input command parsing, dynamic memory management, and system dependent utilities.

**suplib**: A library of routines for reading and writing EXODUS I files, manipulating string, input parsing, and other tasks.

**xhelp**: An interactive and code-callable help system.

**plt**: A medium-level plot library used by fastq, blot, graph, and post.

**svdi**: A low-level device-interface library.

### Scripts

The scripts category is composed of the Unix shell scripts for executing the preprocessing, postprocessing, analysis, and translation codes; and the imake files required to install SEACAS.

### Constitutive Models

Many constitutive models are implemented in the analysis codes.

**Elastic**: Typical linear elastic material using Hooke's Law.

**Elastic-Plastic with Combined Hardening**: Standard von Mises type yield condition with combined kinematic and isotropic linear hardening.

**Viscoplastic**: Simple rate-dependent plasticity.

**Viscoelastic**: Thermoviscoelastic model for glass solidification

**Damage**: Dynamic fracture of brittle rock.

**Soils and Crushable Foam**: Volumetric plasticity model.

**Low Density Foam**: Low density polyurethane foam behavior.

**Hydrodynamic**: Used with Equations of State in PRONTO

**Elastic-Plastic Hydrodynamic**: Combination of elastic-plastic combined hardening with hydrodynamic pressure response.

**Rate and Temperature Dependent Plasticity**: Unified creep plasticity.

**Secondary Creep**: Power hardening steady state creep with elastic bulk response.

**Isotropic Elastic/Plastic**: Power-Law Hardening with Luders Strain.

**Johnson-Cook Strength**: Rate and Temperature-dependent plasticity.

**Hyperelastic**: Stress based on the principal strains.

**Salt Consolidation**: Power hardening steady state creep deviatoric response with consolidation bulk response.

Not all of the constitutive models are implemented in all of the analysis codes. The quasistatic analysis codes typically implement the constitutive models using temperature-dependent properties to facilitate thermal-stress calculations.

## ***SEACAS Abstracts and References***

---

Abstracts and References for selected analysis, preprocessing, postprocessing, translation, and library codes are listed in this section. The codes not listed are primarily translators which are not formally documented.

### ***Algebra***

The ALGEBRA program allows the user to manipulate data from a finite element analysis before it is plotted. The finite element output data is in the form of variable values (e.g., stress, strain, and velocity components) in an EXODUS database. The Algebra program evaluates user-supplied functions of the data and writes the results to an output EXODUS database which can be read by plot programs.

A. P. Gilkey, "ALGEBRA - A Program that Algebraically Manipulates the Output of a Finite Element Analysis (EXODUS Version)," SAND88-1431, Sandia National Laboratories, Albuquerque, New Mexico, August 1988.

### ***Aprepro***

*Aprepro* is an algebraic preprocessor that reads a file containing both general text and algebraic, string, or conditional expressions. It interprets the expressions and outputs them to the output file along with the general text. The syntax used in *Aprepro* is such that all expressions between the delimiters { and } are evaluated and all other text is

simply echoed to the output file. *Aprepro* contains several mathematical functions, string functions, and flow control constructs. In addition, functions are included that, with some additional files, implement a units conversion system and a material database lookup system. *Aprepro* was written primarily to simplify the preparation of parameterized input files for finite element analyses at Sandia National Laboratories; however, it can process any text file that does not use the characters { }.

G. D. Sjaardema, "Aprepro: An Algebraic Pre-processor for Parameterizing Finite Element Analyses," SAND92-2291, Sandia National Laboratories, Albuquerque, New Mexico, December 1992.

### ***Blot***

BLOT is a graphics program for post-processing of finite element analyses output in the EXODUS database format. It is command driven with free-format input and can drive any graphics device supported by the Sandia Virtual Device Interface.

BLOT produces mesh plots with various representations of the analysis output variables. The major mesh plot capabilities are deformed mesh plots, line contours, filled (painted) contours, vector plots of two/three variables (e.g., velocity vectors), and symbol plots of scalar variables (e.g., discrete cracks). Pathlines of analysis variables can also be drawn on the mesh. BLOT's features include element selection by material, element birth and death, multiple views for combining several displays on each plot, symmetry mirroring, and node and element numbering.

BLOT can also produce X-Y curve plots of the analysis variables. BLOT generates time-versus-variable plots or variable-versus-variable plots. It also generates distance-versus-variable plots at selected time steps where the distance is the accumulated distance between pairs of nodes or element centers.

Amy P. Gilkey and John H. Glick, "BLOT - A Mesh and Curve Plot Program for the Output of a Finite Element Analysis," SAND88-1432, Sandia National Laboratories, Albuquerque, New Mexico, August 1991.

R. J. Meyers, "Updates to the postprocessing program BLOT," memo to distribution dated August 21, 1990, Sandia National Laboratories, Albuquerque, NM.

### Conchas

CONCHAS is a linear finite element structural analysis code which is specialized for axisymmetric structures. Loads and responses are limited to those symmetric about a plane which includes the symmetric axis of the structure. CONCHAS will perform eigenanalysis, static analysis, and dynamic analysis. The element library includes consistent-mass shell, solid, and beam elements, nonlinear springs, and concentrated masses. Pre- and postprocessing is available with the separately supported utilities BLOT, PATRAN, and FASTQ.

William C. Mills-Curran and Dennis P. Flanagan, "CONCHAS Users Manual," SAND88-1006, Sandia National Laboratories, Albuquerque, New Mexico, June 1989.

### Cubit

The CUBIT mesh generation environment is a two- and three-dimensional finite element mesh generation tool which is being developed to pursue the goal of robust and unattended mesh generation--effectively automating the generation of quadrilateral and hexahedral elements. It is a solid-modeler based pre-processor that meshes volume and surface solid models for finite element analysis. A combination of techniques including paving, mapping, sweeping, and various other algorithms being developed are available for discretizing the geometry into a finite element mesh. CUBIT also features boundary layer mesh-

ing specifically designed for fluid flow problems. Boundary conditions can be applied to the mesh through the geometry and appropriate files for analysis generated. CUBIT is specifically designed to reduce the time required to create all-quadrilateral and all-hexahedral meshes. The manual is designed to serve as a reference and guide to creating finite element models in the CUBIT environment.

T. D. Blacker, et al, "CUBIT Mesh Generation Environment Volume 1: Users Manual," SAND94-1100, Sandia National Laboratories, Albuquerque, NM, May 1994.

### Exodus I and Genesis File Format

William C. Mills-Curran, Amy P. Gilkey, and Dennis P. Flanagan, "EXODUS: A Finite Element File Format for Pre- and Post-processing," SAND87-2977, Sandia National Laboratories, Albuquerque, New Mexico, September 1988.

L. M. Taylor and D. P. Flanagan and W. C. Mills-Curran, "The GENESIS Finite Element Mesh File Format," SAND86-0910, Sandia National Laboratories, Albuquerque, NM, May 1986.

### Exodus II File Format

EXODUS II is a model developed to store and retrieve data for finite element analyses. It is used for preprocessing (problem definition), postprocessing (results visualization), as well as code to code data transfer. An EXODUS II data file is a random access, machine independent file written and read via C or Fortran library routines which comprise the Application Programming Interface (API). The Network Common Data Form (netCDF, ©1993 University Corporation for Atmospheric Research/Unidata) was selected to handle the low-level data storage for the EXODUS II format.



Larry A. Schoof and Victor R. Yarberr, "EXODUS II: A Finite Element Data Model," SAND92-2137, Sandia National Laboratories, Albuquerque, NM, publication pending.

### ***Fastq***

The FASTQ code is an interactive two-dimensional finite element mesh generation program. It is designed to provide a powerful and efficient tool to both reduce the time required of an analyst to generate a mesh, and to improve the capacity to generate good meshes in arbitrary geometries. It has a number of meshing techniques available: 1) paving for filling arbitrary volumes with elements of arbitrary size, 2) basic mapping techniques, 3) a set of higher-order primitives which have been developed for automatic meshing of commonly encountered shapes (i.e. the triangle, semi-circle, etc.) and conditions (i.e. mesh transitioning from coarse to fine mesh size) and 4) general mesh improvements through smoothing, element reconnection, etc. FASTQ has been designed to allow user flexibility and control. The user interface is built on a layered command level structure. Multiple utilities are provided for input, manipulation, and display of the geometric information, as well as for direct control, adjustment, and display of the generated mesh. Enhanced boundary flagging has been incorporated and multiple element types and output formats are supported. FASTQ includes adaptive meshing capabilities with error estimation, deformed and undeformed remeshing according to the error, element variable remapping, and some basic post-processing plotting.

T. D. Blacker, "FASTQ Users Manual, Version 2.1," SAND88-1326, Sandia National Laboratories, Albuquerque, NM, July 1988.

This paper presents a new mesh generation technique, paving, which meshes arbitrary two-dimensional geometries with an all-quadrilateral mesh. Paving allows varying element size distributions on the boundary as well as the interior of a region. The gener-

ated mesh is well formed (i.e. nearly square elements, elements perpendicular to boundaries, etc.) and geometrically pleasing (i.e. mesh contours tend to follow geometric contours of the boundary). In this paper we describe the theory behind this algorithmic/heuristic technique, evaluate the performance of the approach and present examples of automatically generated meshes.

T. D. Blacker and M. B. Stephenson, "Paving: A New Approach to Automated Quadrilateral Mesh Generation," International Journal for Numerical Methods in Engineering, Volume 32, pp. 811-847, 1991.

### ***Gen3D***

GEN3D is a three-dimensional mesh generation program. The three-dimensional mesh is generated by mapping a two-dimensional mesh into three-dimensions according to one of four types of transformations: translating, rotating, mapping onto a spherical surface, and mapping onto a cylindrical surface. The generated three-dimensional mesh can then be reoriented by offsetting, reflecting about an axis, and revolving about an axis. GEN3D can be used to mesh geometries that are axisymmetric or planar, but, due to three-dimensional loading or boundary conditions, require a three-dimensional finite element mesh and analysis. More importantly, it can be used to mesh complex three-dimensional geometries composed of several sections when the sections can be defined in terms of transformations of two-dimensional geometries. The code GJOIN is then used to join the separate sections into a single body. GEN3D reads and writes two-dimensional and three-dimensional mesh databases in the GENESIS database format; therefore, it is compatible with the pre-processing, postprocessing, and analysis codes used by the Engineering Analysis Department at Sandia National Laboratories, Albuquerque, NM.

Amy P. Gilkey and Gregory D. Sjaardema, "GEN3D: A GENESIS Database 2D to 3D Transformation Program," SAND89-0485, Sandia National Laboratories, Albuquerque, New Mexico, March 1989.

This memo describes the changes that have been made to the GEN3D program since the manual (SAND89-0485) was published. The changes include: (1) New mesh generation options: spline, project, twist, interval, and rotcen transformations; (2) New mesh modification options: change material, change sideset, and change nodeset; (3) New mesh orientation option: scale; and (4) Miscellaneous changes.

G. D. Sjaardema, "Updates to the mesh generation program GEN3D," memo to distribution dated April 11, 1990, Sandia National Laboratories, Albuquerque, New Mexico.

### Genshell

GENSHELL is a three-dimensional shell mesh generation program. The three-dimensional shell mesh is generated by mapping a two-dimensional quadrilateral mesh into three dimensions according to one of several types of transformations: translation, mapping onto a spherical, ellipsoidal, or cylindrical surface, and mapping onto a user-defined spline surface. The generated three-dimensional mesh can then be reoriented by offsetting, reflecting about an axis, revolving about an axis, and scaling the coordinates. GENSHELL can be used to mesh complex three-dimensional geometries composed of several sections when the sections can be defined in terms of transformations of two-dimensional geometries. The code GJOIN is then used to join the separate sections into a single body. GENSHELL updates the EXODUS quality assurance and information records to help track the codes and files used to generate the mesh. GENSHELL reads and writes two-dimensional and three-dimensional mesh databases in the GENESIS database format; there, it is compatible with the pre-processing, postprocessing, and analysis

codes in the Sandia National Laboratories Engineering Analysis Code Access System (SEACAS).

G. D. Sjaardema, "GENSHELL: A Genesis Database 2D to 3D Shell Transformation Program," SAND93-1373, Sandia National Laboratories, Albuquerque, New Mexico, July 1993.

### Gjoin

GJOIN is a two- or three-dimensional mesh combination program. GJOIN combines two or more meshes written in the GENESIS mesh database format into a single GENESIS mesh. Selected nodes in the two meshes that are closer than a specified distance can be combined. The geometry of the mesh databases can be modified by scaling, offsetting, revolving, and mirroring. The combined meshes can be further modified by deleting, renaming, or combining material blocks, sideset identifications, or nodeset identifications. GJOIN is one of the mesh generation tools in the Sandia National Laboratories Engineering Analysis Code Access System (SEACAS). GJOIN is typically used with the other SEACAS mesh generation codes GEN3D, GENSHELL, GREPOS, and Aprepro.

G. D. Sjaardema, "GJOIN: A Program for Merging Two or More GENESIS Databases," SAND92-2290, Sandia National Laboratories, Albuquerque, New Mexico, December 1992

### Grepos

GREPOS is a mesh utility program that repositions or modifies the configuration of a 2D or 3D mesh. Grepos can be used to change the orientation and size of a 2D or 3D mesh; change the material block, nodeset, and sideset IDs; or "explode" the mesh to facilitate viewing of the various parts of the model. Grepos also updates the EXODUS QA and information records to help track the codes and files used to gener-

ate the mesh. GREPOS reads and writes 2D and 3D mesh databases in the GENESIS database format; therefore, it is compatible with the preprocessing, postprocessing, and analysis codes in SEACAS.

G. D. Sjaardema, "GREPOS: A GENESIS Database Repositioning Program," SAND90-0566, Sandia National Laboratories, Albuquerque, New Mexico, April 1990.

### ***Grope***

GROPE is a program that examines the input to a finite element analysis (which is in the GENESIS database format) or the output from an analysis (in the EXODUS database format). Grope allows the user to examine any value in the database. The display can be directed to the user's terminal or to a print file.

G. D. Sjaardema, "GROPE - A GENESIS/EXODUS Database Examination Program," SAND92-2289, Sandia National Laboratories, Albuquerque, NM, October 1992

### ***Jac2D, Jac3D, and Jacq3D***

JAC2D is a two-dimensional finite element program designed to solve quasistatic nonlinear mechanics problems. A set of continuum equations describes the nonlinear mechanics involving large rotation and strain. A nonlinear conjugate gradient method is used to solve the equations. The method is implemented in a two-dimensional setting with various methods for accelerating convergence. Sliding interface logic is also implemented. A four-node Lagrangian uniform strain element is used with hourglass stiffness to control the zero-energy modes. This report documents the elastic and isothermal elastic/plastic material model. Other material models, documented elsewhere, are also available. The program is vectorized for efficient performance on Cray computers. Sample problems described are the bending of a thin beam, the rotation of a

unit cube, and the pressurization and thermal loading of a hollow sphere.

J. H. Biffle and M. L. Blanford, "JAC2D - A Two-Dimensional Finite Element Computer Program for the Nonlinear Quasistatic Response of Solids with the Conjugate Gradient Method," SAND93-1891, Sandia National Laboratories, Albuquerque, New Mexico, May 1994.

JAC3D is a three-dimensional finite element program designed to solve quasistatic nonlinear mechanics problems. A set of continuum equations describes the nonlinear mechanics involving large rotation and strain. A nonlinear conjugate gradient method is used to solve the equations. The method is implemented in a three-dimensional setting with various methods for accelerating convergence. Sliding interface logic is also implemented. An eight-node Lagrangian uniform strain element is used with hourglass stiffness to control the zero-energy modes. This report documents the elastic and isothermal elastic-plastic material models. Other material models, documented elsewhere, are also available. The program is vectorized for efficient performance on CRAY computers. Sample problems described are the bending of a thin beam, the rotation of a unit cube, and the pressurization and thermal loading of a hollow sphere.

J. H. Biffle, "JAC3D - A Three-Dimensional Finite Element Computer Program for the Non-Linear Quasistatic Response of Solids with the Conjugate Gradient Method," SAND87-1305, Sandia National Laboratories, Albuquerque, New Mexico, February 1993.

The nonlinear conjugate gradient procedure is employed in the computer program JACQ3D to solve the steady state and transient nonlinear heat conduction problem for solids in three dimensions. The program can also be used for other types of diffusion problems. The problem is formulated with the finite element method and employs an eight node uniform gradient element with hourglass stiffness to control the zero

energy modes. JACQ3D is highly vectorized to perform efficiently on CRAY computer. Sample problems are included to verify the code and to provide examples of the use of the code.

J. H. Biffle, "JACQ3D - A Three-Dimensional Finite Element Computer Program for Non-Linear Heat Conduction Problems with the Conjugate Gradient Method," In preparation.

### **Merlin II**

The MERLIN II program is designed to transfer data between finite element meshes of arbitrary geometry. The program is structured to accurately interpolate previously computed solutions onto a given mesh and format the resulting data for immediate use in another analysis program. Data from either two-dimensional or three-dimensional meshes may be considered. The theoretical basis and computational algorithms used in the program are described and complete user instructions are presented. Several example problems are included to demonstrate program usage.

D. K. Gartling, "MERLIN II - A Computer Program to Transfer Solution Data Between Finite Element Meshes," SAND89-2989, Sandia National Laboratories, Albuquerque, New Mexico, July 1991.

### **Numbers**

Numbers is a program which reads and stores data from a finite element model described in the EXODUS database format. Within this program are several utility routines which generate information about the finite element model. The utilities currently implemented in Numbers allow the analyst to determine information such as (1) the volume and coordinate limits of each of the materials in the model; (2) the mass properties of the model; (3) the minimum, maximum, and average element volumes for each material; (4) the volume and change in

volume of a cavity; (5) the nodes or elements that are within a specified distance from a user-defined point, line, or plane; (6) an estimate of the explicit central-difference timestep for each material; (7) the validity of contact surfaces or slidelines, that is, whether two surfaces overlap at any point; and (8) the distance between two surfaces.

G. D. Sjaardema, "NUMBERS: A Collection of Utilities for Pre- and Postprocessing Two- and Three-Dimensional EXODUS Finite Element Models," SAND88-0737, Sandia National Laboratories, Albuquerque, New Mexico, March 1989.

### **Pronto2D and Pronto3D**

PRONTO 2D is a two-dimensional transient solid dynamics code for analyzing large deformations of highly nonlinear materials subjected to extremely high strain rates. This Lagrangian finite element program uses an explicit time integration operator to integrate the equations of motion. Four node uniform strain quadrilateral elements are used in the finite element formulation. A number of new numerical algorithms which have been developed for the code are described in this report. An adaptive time step control algorithm is described which greatly improves stability as well as performance in plasticity problems. A robust hourglass control scheme which eliminates hourglass distortions without disturbing the finite element solution is included. All constitutive models in PRONTO are cast in an unrotated configuration defined using the rotation determined from the polar decomposition of the deformation gradient. An accurate incremental algorithm was developed to determine this rotation and is described in detail. A robust contact algorithm was developed which allows for the impact and interaction of deforming contact surfaces of quite general geometry. A number of numerical examples are presented to demonstrate the utility of these algorithms.

Lee M. Taylor and Dennis P. Flanagan, "PRONTO 2D, A Two-Dimensional Transient Solid Dynamics Program," SAND86-0594, Sandia National Laboratories, Albuquerque, New Mexico, March 1987.

PRONTO 3D is a three-dimensional transient solid dynamics code for analyzing large deformations of highly nonlinear materials subjected to extremely high strain rates. This Lagrangian finite element program uses an explicit time integration operator to integrate the equations of motion. Eight-node uniform strain hexahedral elements are used in the finite element formulation. A number of new numerical algorithms which have been developed for the code are described in this report. An adaptive time step control algorithm is described which greatly improves stability as well as performance in plasticity problems. A robust hourglass control scheme which eliminates hourglass distortions without disturbing the finite element solution is included. All constitutive models in PRONTO are cast in an unrotated configuration defined using the rotation determined from the polar decomposition of the deformation gradient. An accurate incremental algorithm was developed to determine this rotation and is described in detail. A robust contact algorithm was developed which allows for the impact and interaction of deforming contact surfaces of quite general geometry. Numerical examples are presented to demonstrate the utility of these algorithms.

L. M. Taylor and D. P. Flanagan, "PRONTO 3D, A Three-Dimensional Transient Solid Dynamics Program," SAND87-1912, Sandia National Laboratories, Albuquerque, New Mexico, March, 1989.

An external code interface is defined which allows other transient applications to communicate with the PRONTO family of finite element programs. This interface is written in ANSI FORTRAN and allows an independent author to specify requirements for an external code to PRONTO. The interface is written such that updates to PRONTO will

not require modifications to the external code.

L. M. Taylor and D. P. Flanagan, "An External Code Interface for the PRONTO Family of Transient Solid Dynamics Programs," SAND87-3003, Sandia National Laboratories, Albuquerque, New Mexico, September, 1988.

PRONTO 2D and PRONTO 3D are two- and three-dimensional solid dynamics codes for analyzing large deformations of highly nonlinear materials subjected to high strain rates. This newsletter is issued to document changes to these codes. As of this writing, the latest version of PRONTO 2D is Version 4.5.6, and the latest version of PRONTO 3D is Version 4.5.6.

This update of the two codes discusses two major modifications to the numerical formulations, three new constitutive models, and the additions and improvements of contact surfaces. Changes in file formats, other miscellaneous revisions, and the availability of PRONTO 2D and PRONTO 3D are also discussed. In addition, updated commands for PRONTO 2D are provided in Appendix A of this newsletter.

S. W. Attaway, "Update of PRONTO 2D and PRONTO 3D Transient Solid Dynamics Program," SAND90-0102, Sandia National Laboratories, Albuquerque, New Mexico, November, 1990.

PRONTO 3D is a three-dimensional transient solid dynamics code for analyzing large deformations of highly nonlinear materials subjected to high strain rates. It is a Lagrangian finite element program with explicit integration of the equations of motion through time. This report documents the implementation of a four-node quadrilateral shell element into Version 6.0 of PRONTO 3D.

This report describes the theory, implementation and use of a four-node shell element. Also described are the required architectural changes made to PRONTO 3D to allow multiple element types. Several test problems are documented for verification of

the PRONTO 3D implementation and for demonstration of computational savings using shell elements for thin structures. These problems also serve as examples for the user. A complete, updated list of the PRONTO 3D input commands is also included.

V. L. Bergmann, "Transient Dynamics Analysis of Plates and Shells with PRONTO 3D," SAND91-1182, Sandia National Laboratories, Albuquerque, New Mexico, September 1991.

This update discusses modifications of PRONTO 3D tailored to the design of fast burst nuclear reactors. A thermoelastic constitutive model and spatially variant thermal history load were added for this special application. Included are descriptions of the thermoelastic constitutive model and the thermal loading algorithm, two example problems used to benchmark the new capability, a user's guide, and PRONTO 3D input files for the example programs. The results from PRONTO 3D thermoelastic finite element analysis are benchmarked against measured data and finite difference calculations.

PRONTO 3D is a three-dimensional transient solid dynamics code for analyzing large deformations of highly non-linear materials subjected to high strain rates. The code modifications are implemented in PRONTO 3D Version 5.3.3.

D. S. Oscar, S. W. Attaway, and J. D. Miller, "Modifications of the PRONTO 3D Finite Element Program Tailored to Fast Burst Nuclear Reactor Design," SAND91-0959, Sandia National Laboratories, Albuquerque, New Mexico, August 1991.

### Sancho

SANCHO is a finite element computer program designed to compute the quasistatic, large deformation, inelastic response of planar or axisymmetric solids. Finite strain constitutive theories for plasticity, volumetric

plasticity, and metallic creep behavior are included. A constant bulk strain, bilinear displacement isoparametric finite element is employed for the spatial discretization. The solution strategy used to generate the sequence of equilibrium solutions is a self-adaptive dynamic relaxation scheme which is based on explicit central difference pseudo-time integration and artificial damping. A master-slave algorithm for sliding interfaces is also implemented. A theoretical development of the appropriate governing equations and a description of the numerical algorithms are presented along with a user's guide which includes several sample problems and their solution.

Charles M. Stone and Raymond D. Krieg and Zelma E. Beisinger, "SANCHO, A Finite Element Computer Program for the Quasistatic, Large Deformation, Inelastic Response of Two-Dimensional Solids," SAND84-2618, Sandia National Laboratories, Albuquerque, New Mexico, April 1985.

### Santos and Santos3D

SANTOS is a finite element computer program designed to compute the quasistatic, large deformation, inelastic response of planar or axisymmetric solids. SANTOS is based on the dynamics program PRONTO2D by L. M. Taylor and D. P. Flanagan. SANTOS utilizes a self-adaptive dynamic relaxation algorithm to achieve a quasistatic solution. The efficiency, speed, through vectorization, and state-of-the-art algorithms that Taylor and Flanagan built into PRONTO2D are maintained in SANTOS. The architecture of the code as well as the user interface is similar for both codes which improves code reliability and encourages the use of both codes by analysts. By utilizing the same material interface, the same constitutive models may be utilized by both codes which allows for coupling of the two codes.

C. M. Stone, documentation not yet available.

### ***Supes Library***

The Software Utilities Package for the Engineering Sciences (SUPES) is a collection of subprograms which perform frequently used non-numerical services for the engineering applications programmer. The three functional categories of SUPES are: (1) input command parsing, (2) dynamic memory management, and (3) system dependent utilities. The subprograms in categories one and two are written in standard FORTRAN-77, while the subprograms in category three are written to provide a standardized FORTRAN interface to several system dependent features.

J. H. Red-Horse, W. C. Mills-Curran, D. P. Flanagan, "SUPES Version 2.1, A Software Utilities Package for the Engineering Sciences," SAND90-0247, Sandia National Laboratories, Albuquerque, New Mexico, May 1990.

### ***Constitutive Models***

B. J. Thorne, "A Damage Model for Rock Fragmentation and Comparison of Calculations With Blasting Experiments in Granite," SAND90-1389, Sandia National Laboratories, Albuquerque, New Mexico, October 1990.

E. P. Chen, "A Computational Model for Jointed Media with Orthogonal Sets of Joints," SAND86-1122, Sandia National Laboratories, Albuquerque, New Mexico, March 1987.

M. K. Neilsen, H. S. Morgan, R. D. Krieg, "A Phenomenological Constitutive Model for Low Density Polyurethane Foams," SAND86-2927, Sandia National Laboratories, Albuquerque, New Mexico, April 1987.

R. S. Chambers, "A Viscoelastic Material Model for Computing Stresses in Glass," SAND90-0645, Sandia National Laboratories, Albuquerque, New Mexico, July 1990.

G. D. Sjaardema, R. D. Krieg, "A Constitutive Model for the Consolidation of WIPP Crushed Salt and Its Use in Analyses of Backfilled Shaft and Drift Configurations," SAND87-1977, Sandia National Laboratories, Albuquerque, New Mexico, October 1987.

C. M. Stone, G. W. Wellman, R. D. Krieg, "A Vectorized Elastic/Plastic Power Law Hardening Material Model Including Lüders Strain," SAND90-0153, Sandia National Laboratories, Albuquerque, New Mexico, March 1990.

J. R. Weatherby, R. D. Krieg, and C. M. Stone, "Incorporation of Surface Tension Into the Structural Finite Element Code SANCHO," SAND89-0509, Sandia National Laboratories, Albuquerque, New Mexico, March 1989.

S. W. Attaway, "A Local Isotropic/Global Orthotropic Finite Element Technique for Modeling the Crush of Wood," SAND88-1449, Sandia National Laboratories, Albuquerque, New Mexico, September 1988.

R. D. Krieg, "A Simple Constitutive Description for Soils and Crushable Foams," SC-DR-72-0883, Sandia National Laboratories, Albuquerque, New Mexico, 1978.

D. J. Bammann, "An Internal Variable Model of Viscoplasticity," *International Journal of Engineering Science*, Volume 22, pp 1041-1053, 1984.

C. M. Stone and G. W. Wellman, "Implementation of Ductile Failure in Pronto2d and Pronto3d," memo to Distribution dated April 8, 1993, Sandia National Laboratories, Albuquerque, New Mexico.

## References

---

<sup>1</sup>William. C. Mills-Curran, Amy P. Gilkey, and Dennis P. Flanagan, “EXODUS: A Finite Element File Format for Pre- and Post-processing,” SAND87-2977, Sandia National Laboratories, Albuquerque, New Mexico, September 1988.

<sup>2</sup>L. M. Taylor and D. P. Flanagan and W. C. Mills-Curran, “The GENESIS Finite Element Mesh File Format,” SAND86-0910, Sandia National Laboratories, Albuquerque, NM, May 1986.

<sup>3</sup>Larry A. Schoof and Victor R. Yarberr, “EXODUSII: A Finite Element Data Model,” SAND92-2137, Sandia National Laboratories, Albuquerque, NM, publication pending.

<sup>4</sup>*American National Standard Programming Language FORTRAN*, American National Standards Institute, Inc., ANSI X3.9-1978, New York, 1978.

<sup>5</sup>*American National Standard for Information Systems--Programming Language C X3.159-1989*. American National Standards Institute, Inc., ANSI X3.159-1989, New York, 1989.

<sup>6</sup>Brian Berliner, “CVS II: Parallelizing Software Development,” Paper presented at the Winter 1990 USENIX Conference, Washington, D.C., 1990.

- 4/22/93: Updated GREPOS description.
- 9/13/93 Added current factual information to the Abstract of this report.
- 9/13/93: Alphabetized abstracts, updated paging.
- 9/13/93: Added GENSHELL abstract.
- 9/13/93: Replaced JAC2D abstract with that of the pending SAND report.
- 9/13/93: Removed all references to SUBWAY.
- 9/13/93: Removed COYOTE and COYOTEII from Figure 2.
- 9/13/93: Added CONCHAS to Figure 2.
- 9/13/93: Occasional editing for standardization of format.
- 6/21/94: Added Exodus II data format information.
- 6/21/94: Added abstracts for CUBIT code and the EXODUS II data format.
- 6/21/94: Added EXODUS translator information.
- 6/21/94: Modified text to define EXODUS file formats where necessary or useful for clarity.
- 9/1/94: Reversed authors of JAS2D Sand Report.
- 9/1/94: Removed extra space, correct typo, page 14
- 9/1/94: Put sentence referring to GENESIS as older form, precursor of EXODUS, in separate paragraph..
- 9/12/94: Added CUBIT to Figure 2.
- 9/12/94: Removed the word GENESIS from Figures 1 and 2.

## Modifications

---

This section lists the changes to this document that have occurred subsequent to publication.

- 4/12/93: Updated JAC3D description to match publication of SAND report.
- 4/12/93: Added ductile failure material model reference by Stone and Wellman.



---

**Modifications**

***Distribution***

			10	1562	G. D. Sjaardema
			1	1832	J. M. Ramage
			1	2565	S. T. Montgomery
			1	6313	J. Jung
			1	6411	A. S. Benjamin
			1	6423	J. F. Dempsey
			1	6513	D. S. Oscar
			1	6522	J. D. Miller
			5	7141	Technical Library
			1	7151	Technical Publications
			10	7613-2	Document Processing for DOE/OSTI
			1	8523-2	Central Technical Files
			6	8741	G. A. Benedetti & staff
			1	8742	M. R. Birnbaum
			1	8742	J. J. Dike
			1	8742	L. I. Weingarten
			5	8743	M. L. Callabresi & staff
1	1400	E. J. Barsis			
1	1401	J. R. Asay			
1	1402	S. S. Dosanjh			
1	1403	G. S. Davidson			
1	1404	J. A. Ang			
13	1425	J. H. Biffle & staff			
50	1425	M. K. Smith			
1	1431	J. M. McGlaun			
1	1431	K. G. Budge			
1	1431	J. S. Peery			
1	1432	W. T. Brown			
1	1433	J. W. Swegle			
15	1434	D. R. Martinez & staff			
1	1500	D. J. McCloskey			
1	1501	C. W. Peterson			
1	1502	P. J. Hommert			
1	1503	L. W. Davison			
1	1504	D. J. McCloskey, actg			
1	1511	J. S. Rottler			
1	1511	D. K. Gartling			
1	1511	M. W. Glass			
1	1511	P. L. Hopkins			
1	1511	M. J. Martinez			
1	1511	P. A. Sackinger			
1	1511	P. R. Schunk			
1	1511	J. D. Zepper			
1	1512	A. C. Ratzel			
1	1513	R. D. Skocypec			
1	1513	R. G. Baca			
1	1513	B. L. Bainbridge			
1	1513	R. E. Hogan, Jr.			
1	1513	J. L. Moya			
1	1551	W. P. Wolfe			
1	1552	C. E. Hailey			
1	1553	W. L. Hermina			
1	1554	W. H. Rutledge			
15	1561	H. S. Morgan & staff			
13	1562	R. K. Thomas & staff			

Second Printing, October 1993

1425 M. K. Smith (50)

Third Printing, August 1994

1503 M. K. Smith (50)