General Information
General Information
Ninth Edition (December 2001)

This major revision replaces GC23-0529-07 and renders it obsolete. This edition applies to the following:

- IBM® Parallel Engineering and Scientific Subroutine Library (Parallel ESSL) for Advanced Interactive Executive (AIX®), program number 5765-C41
- IBM Engineering and Scientific Subroutine Library for AIX (ESSL), program number 5765-C42

Order IBM publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

IBM welcomes your comments. A form for reader’s comments appears at the back of this publication. If the form has been removed, address your comments to:

International Business Machines Corporation
Department 55JA, Mail Station P384
2455 South Road
Poughkeepsie, NY 12601-5400
United States of America

FAX (United States & Canada): 1+845+432-9405
FAX (Other Countries):
Your International Access Code +1+845+432-9405

IBMLink™ (United States customers only): IBMUSM(MHVRCFS)
IBM Mail Exchange: USIB6TC9 at IBMMAIL
Internet e-mail: mhvrdfs@us.ibm.com

If you would like a reply, be sure to include your name, address, telephone number, or FAX number.

Make sure to include the following in your comment or note:
- Title and order number of this book
- Page number or topic related to your comment

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1992, 2001. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
About This Book

This Engineering and Scientific Subroutine Library Products General Information manual is intended to help the customer evaluate IBM Parallel Engineering and Scientific Subroutine Library (Parallel ESSL) for AIX and IBM Engineering and Scientific Subroutine Library for AIX. It contains high-level information about these products, which is helpful in making your purchasing decision.

In this document

Parallel ESSL refers to IBM Parallel Engineering and Scientific Subroutine Library for AIX. ESSL refers to IBM Engineering and Scientific Subroutine Library for AIX.

MPI refers to the Message Passing Interface provided by Parallel Environment (PE).

Abbreviated Names

The abbreviated names used in this book are defined below.

<table>
<thead>
<tr>
<th>Short Name</th>
<th>Full Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX</td>
<td>Advanced Interactive Executive</td>
</tr>
<tr>
<td>BLACS</td>
<td>Basic Linear Algebra Communication Subprograms</td>
</tr>
<tr>
<td>BLAS</td>
<td>Basic Linear Algebra Subprograms</td>
</tr>
<tr>
<td>ESSL</td>
<td>Engineering and Scientific Subroutine Library for AIX</td>
</tr>
<tr>
<td>FDDI</td>
<td>Fiber Distributed Data Interface</td>
</tr>
<tr>
<td>HTML</td>
<td>Hypertext Markup Language</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>IVP</td>
<td>Installation Verification Programs</td>
</tr>
<tr>
<td>LAPACK</td>
<td>Linear Algebra Package</td>
</tr>
<tr>
<td>LAPI</td>
<td>Low-level Application Programming Interface</td>
</tr>
<tr>
<td>MPI</td>
<td>Message Passing Interface</td>
</tr>
<tr>
<td>MPL</td>
<td>Message Passing Library</td>
</tr>
<tr>
<td>PE</td>
<td>Parallel Environment</td>
</tr>
<tr>
<td>PBLAS</td>
<td>Parallel Basic Linear Algebra Subprograms</td>
</tr>
<tr>
<td>PDF</td>
<td>Portable Document Format</td>
</tr>
<tr>
<td>POWER, PowerPC®, POWER2, POWER3, POWER3–II, and POWER4 processors</td>
<td>IBM @server® pSeries™ and RS/6000 processors</td>
</tr>
<tr>
<td>PSSP</td>
<td>Parallel System Support Programs</td>
</tr>
<tr>
<td>ScaLAPACK</td>
<td>Scalable Linear Algebra Package</td>
</tr>
<tr>
<td>SMP</td>
<td>Symmetric Multi-Processing</td>
</tr>
<tr>
<td>SPMD</td>
<td>Single Program Multiple Data</td>
</tr>
<tr>
<td>TLB</td>
<td>Translation Lookaside Buffer</td>
</tr>
<tr>
<td>Short Name</td>
<td>Full Name</td>
</tr>
<tr>
<td>------------</td>
<td>--------------</td>
</tr>
<tr>
<td>US</td>
<td>User Space</td>
</tr>
</tbody>
</table>
Acknowledgement

The Parallel Basic Linear Algebra Subprograms (BLAS), Linear Algebra Equations, and Eigensystem Analysis and Singular Value Analysis routines in Parallel ESSL are based on the Scalable Linear Algebra Package (ScaLAPACK) public domain offering. ScaLAPACK is a scalable linear algebra library for distributed memory concurrent computers. The library was jointly developed by the University of Tennessee, Knoxville, Oak Ridge National Laboratory, and the University of California, Berkeley, and is available from Professor Dongarra, Computer Science Department, the University of Tennessee.
Chapter 1. Engineering and Scientific Subroutine Library (ESSL)—Designed to Meet Your Application Needs

The family of ESSL products consists of:

- Parallel Engineering and Scientific Subroutine Library (Parallel ESSL) for Advanced Interactive Executive (AIX), program number 5765-C41
- Engineering and Scientific Subroutine Library (ESSL) for AIX, program number 5765-C42

These products are state-of-the-art collections of mathematical subroutines that provide a wide range of mathematical functions for many different scientific and engineering applications.

**Parallel ESSL** offers mathematical subroutines in six computational areas (listed on page 4), that run on the following platforms:

- IBM RS/6000® SP™
- Clusters of IBM @server pSeries and RS/6000 workstations

**ESSL** offers mathematical subroutines in nine computational areas (listed on page 5), that run on the following platforms:

- IBM RS/6000 SP

You can use these subroutine libraries to develop and enable many different types of scientific and engineering applications. New applications can be designed and developed to take full advantage of all the capabilities of ESSL. Existing applications can be enabled by replacing comparable subroutines and in-line code with calls to ESSL subroutines. Some of the types of applications that can take advantage of the ESSL capabilities are:

- Structural Analysis
- Computational Chemistry
- Fluid Dynamics Analysis
- Seismic Analysis
- Reservoir Modeling
- Quantitative Analysis
- Time Series Analysis
- Computational Techniques
- Mathematical Analysis
- Dynamic Systems Simulation
- Nuclear Engineering
- Electronic Circuit Design

The Parallel ESSL subroutines can be called from application programs written in Fortran, C, and C++. Parallel ESSL runs under the AIX operating system. On the RS/6000 SP, Parallel System Support Programs (PSSP) is also required.

The ESSL subroutines can be called from application programs written in Fortran, C, and C++. ESSL runs under the AIX operating system.

**Your Choice of Libraries**

This section describes the subroutine libraries available to you.
IBM Parallel Engineering and Scientific Subroutine Library for AIX—Program Number 5765-C41

Parallel ESSL is a scalable mathematical subroutine library that supports parallel processing applications on the IBM RS/6000 SP Systems and clusters of IBM @server pSeries and RS/6000 workstations. Parallel ESSL supports the Single Program Multiple Data (SPMD) programming model using either the Message Passing Interface (MPI) signal handling library or the MPI threaded library. Parallel ESSL provides these run-time libraries:

- The **Parallel ESSL SMP Libraries** are provided for use with the PE MPI threaded library. You may run single or multithreaded applications on all types of nodes; however, you cannot simultaneously call Parallel ESSL from multiple threads. Use these Parallel ESSL libraries if you are using both PE MPI and the Communications Low-level Application Programming Interface (LAPI). The SMP libraries are for use on POWER and PowerPC (for example, POWER3 SMP Thin, Wide, or High Nodes) SMP processors.

  The Parallel ESSL SMP Libraries support both 32-bit environment and 64-bit environment applications.

- The **Parallel ESSL Serial Libraries** are provided for use with the MPI signal handling library. These libraries are tuned for the POWER, POWER3, POWER3-II, POWER4, and PowerPC processors.

Parallel ESSL provides subroutines in six major areas of mathematical computation using high-performance algorithms.

IBM Engineering and Scientific Subroutine Library for AIX—Program Number 5765-C42

ESSL performs computations using high-performance algorithms. ESSL provides these run-time libraries:

- The **ESSL SMP Library** provides thread-safe versions of the ESSL subroutines for use on the SMP (for example, 604e or 630) processors. In addition, a subset of these subroutines are multithreaded versions; that is, they support the shared memory parallel processing programming model. You do not have to change your existing application programs that call ESSL to take advantage of the increased performance of the SMP processors. You can simply re-link your existing programs. For a list of the multithreaded subroutines in the ESSL SMP Library, see the *Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference*.

- The **ESSL Serial Library** provides thread-safe versions of the ESSL subroutines for use on all IBM @server pSeries and RS/6000 processors. You may use this library to develop your own multithreaded applications.

All libraries are designed to provide high levels of performance for numerically intensive computing jobs on these respective processors. All versions provide mathematically equivalent results.

The ESSL libraries support both 32-bit environment and 64-bit environment applications.

---

**Easy to Use**

The **ESSL products have been designed for ease of use**. For example,

- Both products have an easy-to-use call interface and informative error-handling capabilities.
• The subroutines are compatible with public domain subroutine libraries such as Basic Linear Algebra Subprograms (BLAS), Scalable Linear Algebra Package (ScaLAPACK), and Parallel Basic Linear Algebra Subprograms (PBLAS), making it easy to migrate from these libraries.

• You can obtain high performance on SMP processors without requiring extensive knowledge of parallelization techniques.


• The softcopy documentation allows for quick retrieval of information. Hypertext Markup Language (HTML) and Portable Document Format (PDF) versions of the documentation are available on both the product media and the Internet. (See "Obtaining Documentation").

ESSL Internet Resources

This section describes how you can use the ESSL resources available over the Internet.

Obtaining Documentation

To obtain the product documentation in either PDF or HTML format, go to the following IBM Web site:

http://www.ibm.com/eserver/pseries/library

and click on “RS/6000 SP Hardware and Software Books.”

To view the documentation in HTML format, you need access to an HTML browser (such as Netscape Navigator).

To view the documentation in PDF format, you need access to the Adobe Acrobat Reader. The Acrobat Reader is shipped with the AIX Bonus Pack and is also freely available for downloading from the Adobe web site at:

http://www.adobe.com

Accessing ESSL’s Home Pages

The following home page contains information on Parallel ESSL and ESSL:


Getting on the ESSL Mailing List

Late breaking information about ESSL can be obtained by being placed on the ESSL mailing list. Users on the mailing list will receive information about new ESSL function and may receive customer satisfaction surveys and requirements surveys to provide feedback to ESSL Development on the product and user requirements.

You can be placed on the mailing list by sending a request to either of the following, asking to be placed on the ESSL mailing list:

International Business Machines Corporation
ESSL Development
Wide Range of Mathematical Functions

The ESSL libraries provide a variety of complex mathematical functions for many different scientific and engineering applications.

Parallel ESSL Subroutines

The Parallel ESSL mathematical subroutines cover the following areas:

- Level 2 PBLAS
- Level 3 PBLAS
- Linear Algebraic Equations
- Eigensystem Analysis and Singular Value Analysis
- Fourier Transforms
- Random Number Generation
- Utilities

Parallel ESSL subroutines are available in long-precision versions. The Fourier transform subroutines also include short-precision versions.

Level 2 PBLAS include a subset of the standard set of distributed memory parallel versions of the Level 2 BLAS.

Level 3 PBLAS include a subset of the standard set of distributed memory parallel versions of the Level 3 BLAS.

Linear Algebraic Equations Subroutines consist of dense, banded, and sparse subroutines, and include a subset of the ScaLAPACK subroutines.

- Dense Linear Algebraic Equations Subroutines provide:
  - Solutions to linear systems of equations for real and complex general matrices, and their transposes, and for positive definite real symmetric and complex Hermitian matrices.
  - Least squares solutions to linear systems of equations for real and complex general matrices.

- Banded Linear Algebraic Equations Subroutines provide solutions to linear systems of equations for real positive definite symmetric band matrices, real general tridiagonal matrices, diagonally-dominant real general tridiagonal matrices, and real positive definite tridiagonal matrices.
Sparse Linear Algebraic Equations Subroutines and their utility subroutines provide iterative solutions to linear systems of equations for real general sparse matrices.

**Eigensystem Analysis and Singular Value Analysis Subroutines** provide solutions to the algebraic eigensystem analysis problem for real symmetric matrices and complex Hermitian matrices and the real symmetric and complex Hermitian positive definite generalized eigensystem analysis problem. In addition, subroutines to reduce real symmetric and complex Hermitian matrices, real symmetric and complex Hermitian positive definite generalized eigenproblems, and real general matrices to condensed form are provided. These subroutines include a subset of the ScaLAPACK subroutines.

**Fourier Transform Subroutines** perform mixed-radix transforms in two and three dimensions.

**Random Number Generation Subroutine** generates uniformly distributed random numbers.

**Utility Subroutines** perform general service functions, rather than mathematical computations.

**ESSL for AIX**

The ESSL for AIX mathematical subroutines cover the following areas:

- Linear Algebra Subprograms
- Matrix Operations
- Linear Algebraic Equations
- Eigensystem Analysis
- Fourier Transforms, Convolutions and Correlations, and Related Computations
- Sorting and Searching
- Interpolation
- Numerical Quadrature
- Random Number Generation
- Utilities

Several versions of most subroutines are provided, depending on the type of data you are processing. These may include a short- and long-precision real version, a short- and long-precision complex version, and an integer version.

**Linear Algebra Subprograms** consist of vector-scalar, sparse vector-scalar, matrix-vector, and sparse matrix-vector linear algebra subprograms.

- **Vector-Scalar Linear Algebra Subprograms** include a subset of the standard set of Level 1 BLAS and subroutines for other commonly used computations. Both real and complex versions of the subprograms are provided.

- **Sparse Vector-Scalar Linear Algebra Subprograms** operate on sparse vectors; only the nonzero elements of the vectors need to be stored. These subprograms provide functions similar to those of the vector-scalar subprograms and represent a subset of the sparse extensions to the Level 1 BLAS. Both real and complex versions of the subprograms are provided.

- **Matrix-Vector Linear Algebra Subprograms** operate on a higher-level data structure, matrix-vector rather than vector-scalar, using optimized algorithms to improve performance. These subprograms represent a subset of the Level 2 BLAS. Both real and complex versions of the subprograms are provided.
• Sparse Matrix-Vector Linear Algebra Subprograms operate on sparse matrices; only the nonzero elements of the matrix need to be stored. These subprograms provide functions similar to those of the matrix-vector subprograms.

Matrix Operations Subroutines include Level 3 BLAS, as well as the commonly used matrix operations: addition, subtraction, multiplication, and transposition.

Linear Algebraic Equations Subroutines consist of dense, banded, sparse, and linear least squares subroutines.

• Dense Linear Algebraic Equations Subroutines provide solutions to linear systems of equations for real and complex general matrices and their transposes, positive definite real symmetric and complex Hermitian matrices, real symmetric indefinite matrices, and triangular matrices. Some of these subroutines correspond to Level 2, BLAS, Level 3 BLAS, and LAPACK routines.

• Banded Linear Algebraic Equations Subroutines provide solutions to linear systems of equations for real general band matrices, real positive definite symmetric band matrices, real or complex general tridiagonal matrices, real positive definite symmetric tridiagonal matrices, and real or complex triangular band matrices.

• Sparse Linear Algebraic Equations Subroutines provide direct and iterative solutions to linear systems of equations, both for general sparse matrices and their transposes and for sparse symmetric matrices.

• Linear Least Squares Subroutines provide least squares solutions to linear systems of equations for real general matrices. Three methods are provided: one that uses the singular value decomposition; one that uses a QR decomposition with column pivoting; and another that uses a QR decomposition without column pivoting. Some of these subroutines correspond to LAPACK subroutines.

Eigensystem Analysis Subroutines provide solutions to the algebraic eigensystem analysis problem $A z = w z$ and the generalized eigensystem analysis problem $A z = w B z$. These subroutines give you several options for computing eigenvalues or eigenvalues and eigenvectors.

Fourier Transform, Convolution and Correlation, and Related Computation Subroutines are as follows:

• Fourier Transform Subroutines perform mixed-radix transforms in one, two, and three dimensions.

• Convolution and Correlation Subroutines offer a choice between Fourier methods or direct methods. The Fourier-method subroutines contain a high-performance mixed-radix capability. Also, several direct-method subroutines provide decimated output.

• Related Computation Subroutines can be used in general signal processing applications. They are similar to those provided on the IBM 3838 Array Processor; however, the ESSL subroutines generally solve a wider range of problems.

Sorting and Searching Subroutines operate on three types of data: integer, short-precision real, and long-precision real. The sorting subroutines perform a sort with or without index designations. The searching subroutines perform either a binary or a sequential search.

Interpolation Subroutines provide capabilities for polynomial interpolation, local polynomial interpolation, and both one- and two-dimensional cubic spline interpolation.
Numerical Quadrature Subroutines provide one-dimensional methods for integrating a tabulated function and a user-supplied function over a finite, semi-infinite, or infinite region of integration by Gaussian quadrature methods. They also provide a two-dimensional quadrature capability within a rectangular boundary.

Random Number Generation Subroutines generate uniformly or normally distributed random numbers.

Utility Subroutines perform general service functions, rather than mathematical computations.
Chapter 2. Parallel ESSL

The following sections describe key aspects of Parallel ESSL, as well as how to order it.

Highlights of Parallel ESSL

IBM Parallel Engineering and Scientific Subroutine Library (Parallel ESSL) for AIX has the following characteristics:

• Parallel ESSL provides these run-time libraries:
  – The **Parallel ESSL SMP Libraries** are provided for use with the PE MPI threaded library. You may run single or multithreaded applications on all types of nodes. However, you cannot simultaneously call Parallel ESSL from multiple threads. Use these Parallel ESSL libraries if you are using both PE MPI and LAPL. The SMP libraries are for use on the POWER and PowerPC (for example, POWER3–II SMP Thin, Wide, or High Nodes) SMP processors. The Parallel ESSL SMP Libraries support both 32-bit–environment and 64-bit–environment applications.
  – The **Parallel ESSL Serial Libraries** are provided for use with the MPI signal handling library on all types of nodes. These libraries are tuned for the POWER, POWER3, POWER3–II, POWER4, and PowerPC processors.

• Parallel processing subroutines (distributed memory versions) provided in key math areas:
  – Subset of Level 2 and Level 3 Parallel BLAS (PBLAS)
  – Linear Algebraic Equations
    - Subset of ScaLAPACK (dense and banded)
    - Sparse subroutines and their utilities
  – Subset of ScaLAPACK Eigensystem Analysis and Singular Value Analysis
  – Fourier transforms
  – Uniform random number generation

For a list of subroutines, refer to "Chapter 4. Parallel ESSL Subroutines" on page 23.

• Supports the IBM RS/6000 SP and clusters of IBM Eserver pSeries and RS/6000 workstations

• Includes the Basic Linear Algebra Communication Subprograms (BLACS) which provides ease of use for message passing.

• Supports the SPMD programming model:
  – Uses the ESSL subroutines for computations on each processor node
  – Uses the MPI signal handling or threaded library for communication:
    - US—SP Switch, SP Switch2
    - IP—Ethernet, Token Ring, Fiber Distributed Data Interface (FDDI), SP Switch, SP Switch2

• Callable from application programs written in Fortran, C, and C++.

• Provides an environment variable, PESSL_ERROR_SYNC, which allows you to disable error handling synchronization. This may allow you to improve the performance of production-level code.
High Performance of Parallel ESSL

Performance has been the primary objective in the design of the Parallel ESSL subroutines. To achieve this performance goal, the Parallel ESSL subroutines use state-of-the-art algorithms tailored to specific operational characteristics of the hardware. In addition, Parallel ESSL leverages the high performance provided by ESSL, for processor computations.

Choosing a Parallel ESSL Library

The Parallel ESSL library you may use depends on:

1. Your choice of MPI library.

   Note: If you are using LAPI or 64-bit–environment applications, you may only use the MPI threads library.

2. The type of nodes you are running on.

   Table 1. Parallel ESSL Libraries used with the MPI Libraries

<table>
<thead>
<tr>
<th>Node Type</th>
<th>MPI Library</th>
<th>Parallel ESSL Library</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMP¹ or Serial</td>
<td>MPI Signal-Handling Library</td>
<td>Parallel ESSL Serial Library</td>
</tr>
<tr>
<td>SMP¹ or Serial</td>
<td>MPI Threads Library</td>
<td>Parallel ESSL SMP Library²</td>
</tr>
</tbody>
</table>

¹Users may specify multiple user-space MPI tasks per SMP node. For example, you could specify the number of user-space tasks per adapter equal to the number of CPUs in your SMP node.

²If you choose to spawn multiple user-space MPI tasks per SMP node, you may consider explicitly setting the number of threads used by the Parallel ESSL SMP Library per MPI task, by setting the environment variable XLMPLOPTS or OMP_NUM_THREADS. For further details, see the XL Fortran or C for AIX manuals.

Parallel ESSL Techniques

The following techniques are used by most subroutines to optimize performance:

- Minimizing the impact of communications by exchanging larger blocks of data
- Blocking data to match the processor cache size

The design of Parallel ESSL gives you the ability to directly influence performance through the selection of process grid size and block sizes for the distribution of data. While developing and tuning your applications you can select the optimum parameters to suit your problem size. Guidelines for optimum performance and tuning tips are described below:

- Number and types of processors (such as POWER Thin, POWER Wide, POWER2 Thin, POWER3 Wide, POWER4)

  Choosing the number of processors depends primarily on the problem size. It is reasonable to increase the number of processors, if the global problem size increases sufficiently to keep the amount of local data per process at a reasonable size. If, however, using more processes, such as 17 rather than 16, causes you to use a one-dimensional grid rather than a two-dimensional grid, performance may be degraded. See the next item.

- Shape of process grid

  For most subroutines, using a two dimensional (square or as close to square as possible) grid is suggested. For example, if sixteen processors were used, define
a 4 by 4 process grid. For exceptions to this rule, see the subroutine descriptions in the reference section of the Parallel ESSL Version 2 Guide and Reference.

- **Block size(s)**
  The optimal block size in your program depends on the underlying node computations, load balancing, communications, system buffering requirements, problem size, and dimension and shape of the process grid.

- **If you are using the Parallel ESSL SMP Library, your performance may be improved by setting the following environment variables:**
  ```
  export MALLOCMULTIHEAP=true
  export XLSMPOPTS="spins=0:yields=0"
  ```

  **Note:** For details, see the XL Fortran or C for AIX manuals.

- **If you are using the MPI threads library, for a single message-passing thread,**
  specify MP_SINGLE_THREAD=yes to minimize thread overhead.

- **If you are using multiple MPI tasks per node, specify**
  MP_SHARED_MEMORY=yes to specify the use of shared memory (instead of IP or the SP Switch) for message passing between tasks running on the same node.

- **You should be able to improve performance of production-level code by using**
  the PESSL_ERROR_SYNC environment variable to disable error synchronization.

The choices made from the above list generally depend on the specific parallel routine being called.

**Informative Error Handling in Parallel ESSL**

Parallel ESSL features informative and flexible error handling. Exception handling is tailored to the type of error. The Parallel ESSL input-argument checking strategy exceeds the standards defined by existing subroutine library packages. Parallel ESSL checks the validity of the input arguments, and if any of these arguments are invalid, issues an appropriate error message. In addition, Parallel ESSL also provides increased usability by providing a single, comprehensive message when all processors detect the same input argument error.

**Migration Considerations**

This section summarizes the impact of migrating to Parallel ESSL.

**Migrating From Parallel ESSL Version 2.2 to Parallel ESSL Version 2.3:** No changes to your application programs are required if you are migrating from Parallel ESSL Version 2 Release 2 to Parallel ESSL Version 2 Release 3.

**Migrating From Parallel ESSL Version 2.1.2 to Parallel ESSL Version 2.2:** No changes to your application programs are required if you are migrating from Parallel ESSL Version 2 Release 1.2 to Parallel ESSL Version 2 Release 2.

Support has been withdrawn for calling Parallel ESSL from HPF; as a result, the Parallel ESSL HPF libraries, HPF module, HPF IVP, and sample HPF programs are no longer provided.
The Parallel ESSL POWER2 and Thread-Tolerant POWER2 libraries are no longer provided. Existing applications that use these libraries will continue to run because appropriate symbolic links are created at install time to preserve binary compatibility.\(^1\)

However, if you are creating new applications you should use:

- The Parallel ESSL SMP library, instead of the Parallel ESSL POWER2 Thread-Tolerant Library
- The Parallel ESSL Serial Library, instead of the Parallel ESSL POWER2 Library

**Migrating From Parallel ESSL Version 2.1.1 to Parallel ESSL 2.1.2:** The format of the output from PDDTTRF and DTTRF has changed. Therefore, the factorization and solve must be performed using Parallel ESSL Version 2 Release 1.2.

Banded Linear Algebraic Equation subroutines PDPBSV, PDGTSV, PDDTSV and PDPTSV have been modified for the case where N is greater than zero and NRHS is zero so that the matrix is factored. Previously, this was a quick return condition and the matrix was not factored. For all other subroutines, no changes to your application programs are required if you are migrating from Parallel ESSL Version 2 Release 1.1 to Parallel ESSL Version 2 Release 1.2.

**Migrating From Parallel ESSL Version 2 to Parallel ESSL 2.1.1:** The calling sequences for the subroutines in Parallel ESSL Version 2 Release 1 and Parallel ESSL Version 2 Release 1.1 products are identical.

Parallel ESSL provides distinct libraries for AIX 4.2.1 and AIX 4.3.2:

- For AIX 4.2.1, the **Parallel ESSL Thread-Tolerant POWER2 Library** and the **Parallel ESSL SMP Library** were built using the pthreads draft 7 supplied on AIX 4.2.1. This is the same as Parallel ESSL 2.1.
- For AIX 4.3.2, the **Parallel ESSL Thread-Tolerant POWER2 Library** and the **Parallel ESSL SMP Library** were built using the pthreads library that conforms to the IEEE POSIX 1003.1-1996 specification supplied on AIX 4.3.

Applications built using Parallel ESSL 2.1 will continue to run with Parallel ESSL 2.1.1.

**Migrating From Parallel ESSL Version 1 to Parallel ESSL Version 2:** All application programs previously migrated to accommodate the new array descriptor, can run unchanged with Parallel ESSL Version 2 Release 1. However, if you were dependent upon the PESSL_DESC_TYPE environment variable, you must change the array descriptors as defined in the **Parallel ESSL Version 2 Guide and Reference**.

Subroutines with the option of dynamic allocation have been updated to be consistent with ScaLAPACK 1.5. You do not need to update your application programs unless you choose to exploit the new capability. For more information, see the **Parallel ESSL Version 2 Guide and Reference**.

The message-passing and HPF tridiagonal subroutines have been updated to be consistent with ScaLAPACK 1.5. If Parallel ESSL detects a computational error, the value returned in info is the process number where the error occurred. Previously, the index of the pivot where the matrix failed was returned in info. For the

---

1. Customers that require the tuned POWER2 libraries for performance reasons have the option of retaining the Parallel Version 2 Release 1.2 POWER2 libraries when Parallel ESSL Version 2 Release 2 is installed. See the Parallel ESSL Install Memo for details.
message-passing tridiagonal subroutines, the scope of info is now global. You do not have to make any modifications to your existing programs that call these subroutines. For more information, see the appropriate subroutine descriptions in the *Parallel ESSL Version 2 Guide and Reference*.

**Migrating from ScaLAPACK 1.5 to Parallel ESSL Version 2.3:** If you are currently using the ScaLAPACK 1.5 offerings from the Oak Ridge National Laboratory, Parallel ESSL Version 2.3 uses compatible calling sequences with this version of ScaLAPACK. For details, see the *Parallel ESSL Version 2 Guide and Reference*.

**Migrating from a Commercial Parallel Library to Parallel ESSL:** Migrating from a commercial parallel library, such as CMSSL, requires at least a partial, if not substantial, redesign of the application.

**Where to find Details on Migration:** Complete details on migration are given in the "Migrating Your Programs" chapter in the *Parallel ESSL Guide and Reference*.

**Product Requirements**

This section describes the hardware, operating systems, and software products you need when using Parallel ESSL.

**Hardware for Parallel ESSL**

Parallel ESSL runs on IBM RS/6000 SP and clusters of IBM @server pSeries and RS/6000 workstations supported by the operating systems listed under [Operating Systems for Parallel ESSL](#).

64-bit applications require 64-bit hardware.

**Operating Systems for Parallel ESSL**

**Parallel ESSL for AIX** is supported in the following operating system environments:

- AIX 5L™ for POWER Version 5.1, with service (program number 5765-E61)
  - On the SP, you also need the following along with AIX:
    - PSSP for AIX, Version 3.4 or later modification levels (program number 5765-D51)
    - Any additional AIX PTFs required for running on the SP

**Software Products Required for Parallel ESSL**

Parallel ESSL runs with the software products shown in [Table 2](#).

ESSL for AIX must be ordered separately.

**Table 2. Software Products Required for Use with Parallel ESSL**

<table>
<thead>
<tr>
<th>For Compiling</th>
<th>For Linking, Loading, or Running</th>
</tr>
</thead>
<tbody>
<tr>
<td>XL Fortran for AIX, Version 7.1.1 (program number 5765-E02)</td>
<td>XL Fortran Run-time Environment for AIX, Version 7.1.1 (program number 5765-E03)</td>
</tr>
<tr>
<td>C for AIX, Version 5.0.2 (program number 5765-E32)</td>
<td>Parallel Environment for AIX, Version 3.2 (program number 5765-D93)</td>
</tr>
<tr>
<td>VisualAge® C++ Professional for AIX Version 5.0.2 (program number 5765-E26)</td>
<td>ESSL for AIX, Version 3.3 (program number 5765-C42)</td>
</tr>
<tr>
<td></td>
<td>C libraries¹</td>
</tr>
</tbody>
</table>

¹ AIX includes the C libraries and math libraries in the Application Development Toolkit.
Thread Safety

Parallel ESSL is not thread safe; however, Parallel ESSL is thread tolerant and can therefore be called from a single thread of a multithreaded application. Multiple simultaneous calls to Parallel ESSL from different threads of a single process causes unpredictable results.

For more information on thread programming concepts, see IBM AIX General Programming Concepts: Writing and Debugging Programs.

Installation and Customization Requirements

Parallel ESSL is distributed on a CD. The Parallel Engineering and Scientific Subroutine Library for AIX Version 2 Installation Memo provides the detailed information you need to install Parallel ESSL on AIX.

The Parallel ESSL product is packaged in accordance with the AIX guidelines. The product can be installed using the smit command, as described in the IBM Parallel System Support Programs for AIX: Administration Guide. The product can be installed on multiple nodes using the dsh command, as described in the IBM Parallel System Support Programs for AIX: Administration Guide and the installp command, as described in the IBM AIX Commands Reference.

Online Documentation Requirements

The Parallel Engineering and Scientific Subroutine Library for AIX Version 2 Guide and Reference is available in PDF and HTML format on the product media. To find out the location of these files on your system, contact your system administrator or installer.

The Parallel Engineering and Scientific Subroutine Library for AIX Version 2 Guide and Reference is also available in PDF and HTML format on the Internet. (See “Obtaining Documentation” on page 3.)

Ordering Parallel ESSL

You can order Parallel ESSL from your IBM Marketing Representative, or by:
• calling 1-800-IBM-CALL (1-800-426-2255)
• using the IBM Web site at:
  http://www.ibm.com/shop/

Ask for:
• IBM Parallel Engineering and Scientific Subroutine Library for AIX (program number 5765-C41)

Parallel ESSL Product Package

The Parallel ESSL package provides:
• Parallel ESSL libraries
• Parallel ESSL header file for C and C++
• Parallel ESSL modules
• Installation Verification Programs
• Parallel ESSL Documentation
  – Softcopy:

- Print-on-Demand:

- Printed:
  - Parallel ESSL Version 2 Licensed Information, GA22-7303
  - Parallel ESSL Proof of Entitlement
  - Parallel ESSL Version 2 Installation Memo, GI10-0607
  - Parallel ESSL Memo to New Users, GI10-0606

The publication Parallel ESSL Version 2 Guide and Reference, SA22-7273, is available in softcopy only.
Chapter 3. ESSL for AIX

The following sections describe key aspects of the ESSL for AIX product, as well as how to order it.

Highlights of ESSL for AIX

- ESSL provides these run-time libraries:
  - The **ESSL SMP Library** provides thread-safe versions of the ESSL subroutines for use on the SMP (for example, 604e or 630) processors. In addition, a subset of these subroutines are multithreaded versions; that is, they support the shared memory parallel processing programming model. You do not have to change your existing application programs that call ESSL to take advantage of the increased performance of the SMP processors. You can simply re-link your existing application programs. For a list of the multithreaded subroutines in the ESSL SMP Library, see *Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference*.
  - The **ESSL Serial Library** provides thread-safe versions of the ESSL subroutines for use on all IBM @server pSeries and RS/6000 processors. You may use this library to develop your own multithreaded applications.

All libraries are designed to provide high levels of performance for numerically intensive computing jobs on these respective processors. All versions provide mathematically equivalent results.

The ESSL libraries support both 32-bit environment and 64-bit environment applications. For complete details, see *Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference*, available on the following IBM Web site:

http://www.ibm.com/eserver/pseries/library

- Callable from Fortran, C, and C++ programs.
- For a list of ESSL subroutines see “Chapter 5. ESSL for AIX Subroutines” on page 27.

High Performance of ESSL for AIX

**Algorithms:** The ESSL subroutines have been designed to provide high performance. To achieve this performance, the subroutines use state-of-the-art algorithms tailored to specific operational characteristics of the hardware, such as cache size, Translation Lookaside Buffer (TLB) size, and page size.

Most subroutines use the following techniques to optimize performance:
- Managing the cache and TLB efficiently so the hit ratios are maximized; that is, data is blocked so it stays in the cache or TLB for its computation.
- Accessing data stored contiguously—that is, using stride-1 computations.
- Exploiting the large number of available floating-point registers.
- Using algorithms that minimize paging.
- On the SMP processor:
  - The ESSL SMP Library is designed to exploit the processing power and shared memory of the SMP processor. In addition, a subset of the ESSL SMP
subroutines have been coded to take advantage of increased performance from multithreaded (parallel) programming techniques. For a list of the multithreaded subroutines in the ESSL SMP Library, see the ESSL Version 3 Guide and Reference. Many of the other ESSL SMP subroutines make one or more calls to multithreaded subroutines, and also benefit from increased performance.

Choosing the number of threads depends on the problem size, the specific subroutine being called, and the number of physical processors you are running on. To achieve optimal performance, experimentation is necessary; however, picking the number of threads equal to the number of online processors generally provides good performance in most cases. In some cases, performance may increase if you choose the number of threads to be less than the number of online processors.

You should use either the XL Fortran XLSMPOPTS or the OMP_NUM_THREADS environment variable to specify the number of threads you want to create.

- On the POWER processor:
  - Using algorithms that balance floating-point operations with loads in the innermost loop.
  - Using algorithms that minimize stores in the innermost loops.
  - Structuring the ESSL subroutines so, where applicable, the compiled code uses the Multiply-Add instructions. Neglecting overhead, these instructions perform two floating-point operations per cycle.

- On the POWER3 or POWER3–II processor:
  - Structuring the ESSL subroutines so, where applicable, the compiled code fully utilizes the dual floating-point execution units. Because two Multiply-Add instructions can be executed each cycle, neglecting overhead, this allows four floating-point operations per cycle to be performed.
  - Structuring the ESSL subroutines so, where applicable, the compiled code takes full advantage of the hardware data prefetching.

- On the POWER4 processor:
  - Structuring the ESSL subroutines so, where applicable, the compiled code fully utilizes the dual floating-point execution units. Because two Multiply-Add instructions can be executed each cycle, neglecting overhead, this allows four floating-point operations per cycle to be performed.
  - Structuring the ESSL subroutines so, where applicable, the compiled code takes full advantage of the hardware data prefetching.

Mathematical Techniques: All areas of ESSL use state-of-the-art mathematical techniques to achieve high performance. For example, the matrix-vector linear algebra subprograms operate on a higher-level data structure, matrix-vector rather than vector-scalar. As a result, they optimize performance directly for your program and indirectly through those ESSL subroutines using them.

Migration Considerations

This section summarizes the impact of migrating to ESSL.

Migrating From ESSL Version 3 Release 2 to ESSL Version 3 Release 3: No changes to your FORTRAN or C application programs are required if you are migrating from ESSL Version 3 Release 2 to ESSL Version 3 Release 3.
Changes may be required in your C++ application programs. The ESSL Version 3 Release 2 header file inconsistently declared output scalar arguments in the function prototypes. Some were declared to be type reference and some as pointers. In ESSL Version 3 Release 3, by default, all scalar output arguments are declared to be type reference; optionally, you may choose to have all scalar output arguments declared as pointers. (For more detailed information, see Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference.)

If you used any of the following subroutines in your C++ application program, you will need to make updates to your application program and/or compilation procedures:

- _ROTG
- _GEFCD
- _PPFCD
- _POFCD
- _PPICD
- _POICD
- _GELLS
- _PINT
- _CSINT
- _PTNQ
- _URAND
- _NRAND
- _URXOR
- STRIDE
- DSRSM

AIX 5L for POWER Version 5.1, with service, provides a more scalable application binary interface for 64-bit applications. To take advantage of the scalability improvements to 64-bit programs, all 64-bit applications and libraries must be recompiled on AIX 5L for POWER Version 5.1, with service.

**Migrating From ESSL Version 3 Release 1.2 to ESSL Version 3 Release 2:** The calling sequences for the subroutines in the ESSL Version 3 Release 1.2 and the ESSL Version 3 Release 2 product are identical. No changes to your application programs are required if you are migrating from ESSL Version 3 Release 1.2 to ESSL Version 3 Release 2.

The ESSL POWER and Thread-Safe POWER libraries have been replaced by a thread-safe library referred to as the ESSL Serial Library.

The ESSL POWER2 and Thread-Safe POWER2 libraries are no longer provided. Existing applications that use these libraries will continue to run because appropriate symbolic links are created at install time to preserve binary compatibility. However, if you are creating new applications you should use:

- The ESSL Serial or SMP library instead of the ESSL Thread-Safe POWER2 Library.
- The ESSL Serial Library instead of the ESSL POWER2 Library.

**Migrating From ESSL Version 3 Release 1.1 to ESSL Version 3 Release 1.2:** The calling sequences for the subroutines in the ESSL Version 3 Release 1.1 and the

---

2. Customers that require the tuned POWER2 libraries for performance reasons have the option of retaining the Version 3 Release 1.2 POWER2 libraries when ESSL Version 3 Release 2 is installed. See the ESSL Install Memo for details.
ESSL Version 3 Release 1.2 product are identical. No changes to your application programs are required if you are migrating from ESSL Version 3 Release 1.1 to ESSL Version 3 Release 1.2.

**Migrating From ESSL Version 3 to ESSL Version 3 Release 1.1:** The calling sequences for the subroutines in the ESSL Version 3 Release 1 and the ESSL Version 3 Release 1.1 product are identical.

Distinct Libraries are provided for AIX 4.2.1 and AIX 4.3.2:
- For AIX 4.2.1, the ESSL Thread-Safe Library, the ESSL Thread-Safe POWER2 Library, and the ESSL SMP Library were built using the pthreads draft 7 supplied on AIX 4.2.1. This is the same as ESSL 3.1.
- For AIX 4.3.1, the ESSL Thread-Safe Library, the ESSL Thread-Safe POWER2 Library, and the ESSL SMP Library were built using the pthreads library that conforms to the IEEE POSIX 1003.1-1996 specification supplied on AIX 4.3.

Applications built using ESSL 3.1 will continue to run with ESSL 3.1.1.

If you are migrating to a 64-bit environment you may need to make changes to your calls to ERRSET.

**Migrating From ESSL Version 2 to ESSL Version 3:** The calling sequences for the subroutines in the ESSL Version 2 and ESSL Version 3 product are identical. This includes the new ESSL SMP and Thread-Safe Libraries that are included in the ESSL Version 3 product. You do not have to change your existing application programs that call ESSL when migrating to the ESSL Version 3 product. You must, however, re-link your application program.

ESSL messages have been reformatted and are now in a message catalog. Some input-argument and computational error message numbers were changed to attention message numbers. The old message numbers can still be used when calling ERRSET, however, you should migrate to the new message numbers. For details, see the *Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference*. For the _GEF and _GEFCD subroutines, the first column of the matrix L with the corresponding \( U_{ii} = 0 \) diagonal element is identified in a computational error message. Previously, the last column was identified. You do not have to make any modifications to your existing application programs that call these subroutines.

**Migrating from LAPACK:** ESSL contains a few subroutines that conform to the LAPACK interface, see Appendix B in the *Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference*. If you are using these subroutines, no coding changes are needed to migrate to ESSL.

Additionally, you may be interested in using the Call Conversion Interface (CCI) that is available with LAPACK. The CCI substitutes a call to an ESSL subroutine in place of an LAPACK subroutine whenever an ESSL subroutine provides either functional or near-functional equivalence. Using the CCI allows LAPACK users to obtain the optimized performance of ESSL for an additional subset of LAPACK subroutines. For details, see *Call Conversion Interface (CCI) for LAPACK/ESSL*, LAPACK Working Note 82, Department of Computer Science University of Tennessee, Knoxville, Tennessee. (You can download this document from: [http://www.netlib.org/lapack/lawns/lawn82.ps](http://www.netlib.org/lapack/lawns/lawn82.ps))
Migrating Between ESSL/370 and ESSL Version 3: The ESSL calling sequences are predominantly compatible with the ESSL/370 calling sequences; therefore, most ESSL mainframe application programs can be easily ported.

Where to Find Details on Migration: Complete details on migration are given in the “Migrating Your Programs” chapter of the Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference manual.

Product Requirements

This section describes the hardware and software products you need when using ESSL.

Hardware for ESSL

ESL runs on IBM @server pSeries and RS/6000 processors supported by the AIX operating system(s) listed under “Operating Systems for ESSL.”

64-bit applications require 64-bit hardware.

Operating Systems for ESSL

ESL is supported in the following operating system environments:

- AIX 5L for POWER Version 5.1, with service (program number 5765-E61)

Software Products Required for ESSL

ESL runs with the software products shown below.

Table 3. Software Products Required for Use with ESSL

<table>
<thead>
<tr>
<th>For Compiling</th>
<th>For Linking, Loading, or Running</th>
</tr>
</thead>
<tbody>
<tr>
<td>XL Fortran for AIX, Version 7.1.1 (program number 5765-E02) –or– VisualAge C++ Professional for AIX Version 5.0.2 –or– C for AIX, Version 5.0.2</td>
<td>XL Fortran Run-Time Environment for AIX, Version 7.1.1 (program number 5765-E03) –and– C libraries¹</td>
</tr>
</tbody>
</table>

¹ The AIX product includes the C and math libraries in the Application Development Toolkit.

Thread Safety

ESL provides thread-safe versions of the ESSL subroutines for use on all IBM @server pSeries and RS/6000 processors. For more information, see the Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference.

For more information on thread programming concepts, see IBM AIX General Programming Concepts: Writing and Debugging Programs.

Installation and Customization Requirements

ESL: The ESSL licensed program is distributed on a CD. The Engineering and Scientific Subroutine Library Version 3 Installation Memo, provides the detailed information you need to install ESSL on AIX.

The ESSL product is packaged in accordance to the AIX guidelines, as described in the IBM AIX General Programming Concepts: Writing and Debugging Programs.
product can be installed using the `smit` command, as described in the *IBM AIX System Management Guide: Operating System and Devices*.

### Online Documentation Requirements

The *Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference* is available in PDF and HTML format on the product media. To find out the location of these files on your system, contact your system administrator or installer.

The *Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference* is also available in PDF and HTML format on the Internet. (See “Obtaining Documentation” on page 3.)

### Ordering ESSL for AIX

You can order ESSL from your IBM Marketing Representative, or by:
- calling 1-800-IBM-CALL (1-800-426-2255)
- using the IBM Web site at:
  
  http://www.ibm.com/shop/

Ask for the ESSL product:
- IBM Engineering and Scientific Subroutine Library for AIX (program number 5765-C42)

### ESSL Product Package

Each ESSL product package provides:
- ESSL libraries
- ESSL header file for C and C++
- Installation Verification Programs
- ESSL for AIX Documentation
  - Softcopy:
  - Print-on-Demand:
  - Printed:
    - *ESSL Version 3 Licensed Information*, GA22-7302
    - ESSL Proof of Entitlement
    - *Engineering and Scientific Subroutine Library Version 3 Installation Memo*, GI10-0604
    - *ESSL Memo to New Users*, GI10-0603

The publication *Engineering and Scientific Subroutine Library for AIX Version 3 Guide and Reference*, SA22-7272 is available in softcopy only.
## Chapter 4. Parallel ESSL Subroutines

This chapter includes a complete list of the subroutines offered in Parallel ESSL Version 2 Release 3.

The subroutines are grouped by mathematical area, where:

Φ A capital Φ indicates the subroutines that are new for Parallel ESSL Version 2 Release 3.

The subroutine names are distinguished by a prefix based on the following letters:

- P for Parallel ESSL message passing subroutines (first letter)
- S for short-precision real
- D for long-precision real
- Z for long-precision complex

**Note:** Level 2 PBLAS, Level 3 PBLAS, Dense and Banded Linear Algebraic Equations, and Eigensystem Analysis and Singular Value Analysis subroutines were designed in accordance with the proposed ScaLAPACK standard. If these subroutines do not comply with the standard as approved, IBM will consider updating them to do so. If IBM updates these subroutines, the update could require modifications of the calling application program.

Following is a complete list of the subroutines offered in Parallel ESSL.

### Level 2 PBLAS

<table>
<thead>
<tr>
<th>Description</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matrix-Vector Product for a General Matrix or Its Transpose</td>
<td>PDGEMV</td>
</tr>
<tr>
<td>Matrix-Vector Product for a Real Symmetric or a Complex Hermitian Matrix</td>
<td>PDSYMV</td>
</tr>
<tr>
<td>Rank-One Update of a General Matrix</td>
<td>PDGER</td>
</tr>
<tr>
<td>Rank-One Update of a Real Symmetric or a Complex Hermitian Matrix</td>
<td>PDSYR</td>
</tr>
<tr>
<td>Rank-Two Update of a Real Symmetric or a Complex Hermitian Matrix</td>
<td>PDSYR2</td>
</tr>
<tr>
<td>Matrix-Vector Product for a Triangular Matrix or Its Transpose</td>
<td>PDTRMV</td>
</tr>
<tr>
<td>Solution of Triangular System of Equations with a Single Right-hand Side</td>
<td>PDTRSV</td>
</tr>
</tbody>
</table>

### Level 3 PBLAS

<table>
<thead>
<tr>
<th>Description</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matrix-Matrix Product for a General Matrix or Its Transpose or Its Conjugate Transpose</td>
<td>PDGEMM, PZGEMM</td>
</tr>
<tr>
<td>Matrix-Matrix Product Where One Matrix is Real Symmetric or Complex Hermitian</td>
<td>PDSYMM, PZSYMM</td>
</tr>
<tr>
<td>Triangular Matrix-Matrix Product</td>
<td>PDTRMM, PZTRMM</td>
</tr>
</tbody>
</table>
### Linear Algebraic Equations

#### Dense Linear Algebraic Equations

<table>
<thead>
<tr>
<th>Task</th>
<th>Routine</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Matrix Factorization and Solve</td>
<td>PDGESV</td>
</tr>
<tr>
<td>General Matrix Factorization</td>
<td>PDGETRF</td>
</tr>
<tr>
<td>General Matrix Solve</td>
<td>PDGERS</td>
</tr>
<tr>
<td>General Matrix Inverse</td>
<td>PDGETRI</td>
</tr>
<tr>
<td>Estimate the Reciprocal of the Condition Number of a General Matrix</td>
<td>PDGECON</td>
</tr>
<tr>
<td>General Matrix QR Factorization</td>
<td>PDGEQRF</td>
</tr>
<tr>
<td>General Matrix Least Squares Solution</td>
<td>PDGELS</td>
</tr>
<tr>
<td>Positive Definite Real Symmetric or Complex Hermitian Matrix Factorization and Solve</td>
<td>PDPOSV</td>
</tr>
<tr>
<td>Positive Definite Real Symmetric or Complex Hermitian Matrix Solve</td>
<td>PDPOTRS</td>
</tr>
<tr>
<td>Positive Definite Real Symmetric or Complex Hermitian Matrix Factorization</td>
<td>PDPOTRA</td>
</tr>
</tbody>
</table>

#### Banded Linear Algebraic Equations

<table>
<thead>
<tr>
<th>Task</th>
<th>Routine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive Definite Symmetric Band Matrix Factorization and Solve</td>
<td>PDPBSV</td>
</tr>
<tr>
<td>Positive Definite Symmetric Band Matrix Factorization</td>
<td>PDPBTRF</td>
</tr>
<tr>
<td>Positive Definite Symmetric Band Matrix Solve</td>
<td>PDPBTRS</td>
</tr>
<tr>
<td>General Tridiagonal Matrix Factorization and Solve</td>
<td>PDGT5V</td>
</tr>
<tr>
<td>General Tridiagonal Matrix Factorization</td>
<td>PDGTTRF</td>
</tr>
<tr>
<td>General Tridiagonal Matrix Solve</td>
<td>PDGTTRS</td>
</tr>
<tr>
<td>Diagonally-Dominant General Tridiagonal Matrix Factorization and Solve</td>
<td>PDDBTSV</td>
</tr>
<tr>
<td>Diagonally-Dominant General Tridiagonal Matrix Factorization</td>
<td>PDDBTTRF</td>
</tr>
<tr>
<td>Diagonally-Dominant General Tridiagonal Matrix Solve</td>
<td>PDDBTTRS</td>
</tr>
<tr>
<td>Positive Definite Symmetric Tridiagonal Matrix Factorization and Solve</td>
<td>PDPTSV</td>
</tr>
</tbody>
</table>
Positive Definite Symmetric Tridiagonal Matrix Factorization  
Positive Definite Symmetric Tridiagonal Matrix Solve

**Fortran 90 Sparse Linear Algebraic Equations and Utilities**

- Allocates Space for an Array Descriptor for a General Sparse Matrix: PADALL
- Allocates Space for a General Sparse Matrix: PSPALL
- Allocates Space for a Dense Vector: PGEALL
- Inserts Local Data into a General Sparse Matrix: PSPINS
- Inserts Local Data into a Dense Vector: PGEINS
- Assembles a General Sparse Matrix: PSPASB
- Assembles a Dense Vector: PGEASB
- Preconditioner for a General Sparse Matrix: PSPGPR
- Iterative Linear System Solver for a General Sparse Matrix: PSPGIS
- Deallocates Space for a Dense Vector: PGEFREE
- Deallocates Space for a General Sparse Matrix: PSPFREE
- Deallocates Space for an Array Descriptor for a General Sparse Matrix: PADFREE

**Fortran 77 Sparse Linear Algebraic Equations and Utilities**

- Initializes an Array Descriptor for a General Sparse Matrix: PADINIT
- Initializes a General Sparse Matrix: PDSINIT
- Inserts Local Data into a General Sparse Matrix: PDSINS
- Inserts Local Data into a Dense Vector: PDGEINS
- Assembles a General Sparse Matrix: PDSASB
- Assembles a Dense Vector: PDGEASB
- Preconditioner for a General Sparse Matrix: PDSPGPR
- Iterative Linear System Solver for a General Sparse Matrix: PDSPGIS

**Eigensystem Analysis and Singular Value Analysis**

- Selected Eigenvalues and, Optionally, the Eigenvectors of a Real Symmetric or Complex Hermitian Matrix: PDSYEVX
- Selected Eigenvalues and, Optionally, the Eigenvectors of a Real Symmetric or Complex Hermitian Positive Definite Generalized Eigenproblem: PZHEEVX
- Reduce a Real Symmetric or Complex Hermitian Matrix to Tridiagonal Form: PDSYTRD
- Reduce a Real Symmetric or Complex Hermitian Positive Definite Generalized Eigenproblem to Standard Form: PZHEGST
- Reduce a General Matrix to Upper Hessenberg Form: PDGEHRD
- Reduce a General Matrix to Bidiagonal Form: PDGEBRD

**Fourier Transforms**

- Complex Fourier Transform in Two Dimensions: PSCFT2, PDCFT2
### Random Number Generation

<table>
<thead>
<tr>
<th>Description</th>
<th>Library</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform Random Number Generator</td>
<td>PDURNG</td>
</tr>
</tbody>
</table>

### Utilities

<table>
<thead>
<tr>
<th>Description</th>
<th>Library</th>
</tr>
</thead>
<tbody>
<tr>
<td>Determine the Level of Parallel ESSL Installed on Your System</td>
<td>IPESL</td>
</tr>
<tr>
<td>Compute the Number of Rows or Columns of a Block-Cyclically Distributed Matrix Contained in a Process</td>
<td>NUMROC</td>
</tr>
<tr>
<td>General Matrix Norm</td>
<td>PDLANGE</td>
</tr>
<tr>
<td></td>
<td>PZLANGE</td>
</tr>
</tbody>
</table>
Chapter 5. ESSL for AIX Subroutines

This chapter includes a complete list of the subroutines offered in ESSL Version 3 Release 3.

The subroutines are grouped by mathematical area, where:

- A capital phi indicates the subroutines that are new for ESSL Version 3 Release 3.
- A box indicates the subroutines that correspond to Level 1 BLAS.
- A diamond indicates the subroutines that correspond to Level 2 BLAS.
- A left arrowhead indicates the subroutines that correspond to Level 3 BLAS.
- A double dagger indicates the subroutines that correspond to LAPACK.
- A section symbol indicates the subroutines that are provided for migration from earlier releases of ESSL and are not intended for use in new programs.

Both short- and long-precision real versions of the subroutines are provided in most areas of ESSL. In some areas, short- and long-precision complex versions are also provided, and, occasionally, an integer version is provided. The subroutine names are distinguished by a one-, two-, or three-letter prefix based on the following letters:

- S for short-precision real
- D for long-precision real
- C for short-precision complex
- Z for long-precision complex
- I for integer

Some of the linear algebra subprograms, matrix operations subroutines, and linear algebraic equation subroutines were designed in accordance with the Level 1, Level 2, and Level 3 BLAS, and LAPACK de facto standard. If these subprograms do not comply with the standard as approved, IBM will consider updating them to do so. If IBM updates these subprograms, the updates could require modifications of the calling application program.

Linear Algebra Subprograms

Vector-Scalar Linear Algebra Subprograms

Position of the First or Last Occurrence of the Vector Element Having the Largest Magnitude IDAMAX
Position of the First or Last Occurrence of the Vector Element Having Minimum Absolute Value IDAMIN
Position of the First or Last Occurrence of the Vector Element Having Maximum Value IDMAX
Position of the First or Last Occurrence of the Vector Element Having Minimum Value IDMIN
<table>
<thead>
<tr>
<th>Description</th>
<th>SASUM</th>
<th>DASUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum of the Magnitudes of the Elements in a Vector</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiply a Vector $X$ by a Scalar, Add to a Vector $Y$, and Store in the Vector $Y$</td>
<td>SASPY</td>
<td>DASPY</td>
</tr>
<tr>
<td>Copy a Vector</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dot Product of Two Vectors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Compute SAXPY or DAXPY N Times</td>
<td>SNAXPY</td>
<td>DNAXPY</td>
</tr>
<tr>
<td>Compute Special Dot Products N Times</td>
<td>SNDOT</td>
<td>DNDOT</td>
</tr>
<tr>
<td>Euclidean Length of a Vector with Scaling of Input to Avoid Destructive Underflow and Overflow</td>
<td>SCNRM2*</td>
<td>DZNRM2*</td>
</tr>
<tr>
<td>Euclidean Length of a Vector with No Scaling of Input</td>
<td>SNORM2</td>
<td>DNORM2</td>
</tr>
<tr>
<td>Construct a Givens Plane Rotation</td>
<td>SROTG</td>
<td>DROTG</td>
</tr>
<tr>
<td>Apply a Plane Rotation</td>
<td>SROT</td>
<td>DROT</td>
</tr>
<tr>
<td>Multiply a Vector $X$ by a Scalar and Store in the Vector $X$</td>
<td>SSCAL</td>
<td>DSCAL</td>
</tr>
<tr>
<td>Interchange the Elements of Two Vectors</td>
<td>SSWAP</td>
<td>DSWAP</td>
</tr>
<tr>
<td>Add a Vector $X$ to a Vector $Y$ and Store in a Vector $Z$</td>
<td>SVEA</td>
<td>DVEA</td>
</tr>
<tr>
<td>Subtract a Vector $Y$ from a Vector $X$ and Store in a Vector $Z$</td>
<td>SVES</td>
<td>DVES</td>
</tr>
<tr>
<td>Multiply a Vector $X$ by a Vector $Y$ and Store in a Vector $Z$</td>
<td>SYEM</td>
<td>DYM</td>
</tr>
<tr>
<td>Multiply a Vector $X$ by a Scalar and Store in a Vector $Y$</td>
<td>SCAL</td>
<td>ZCAL</td>
</tr>
<tr>
<td>Multiply a Vector $X$ by a Scalar, Add to a Vector $Y$, and Store in a Vector $Z$</td>
<td>SDDOT</td>
<td>ZDDOT</td>
</tr>
</tbody>
</table>

### Sparse Vector-Scalar Linear Algebra Subprograms

<table>
<thead>
<tr>
<th>Description</th>
<th>SSCTR</th>
<th>DSCTR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scatter the Elements of a Sparse Vector $X$ in Compressed-Vector Storage Mode into Specified Elements of a Sparse Vector $Y$ in Full-Vector Storage Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gather Specified Elements of a Sparse Vector $Y$ in Full-Vector Storage Mode</td>
<td>SGTHR</td>
<td>DGTHR</td>
</tr>
<tr>
<td>Full-Vector Storage Mode into a Sparse Vector $X$ in Compressed-Vector Storage Mode</td>
<td>CGTHR</td>
<td>ZGTHR</td>
</tr>
<tr>
<td>Gather Specified Elements of Sparse Vector $Y$ in Full-Vector Mode into Sparse Vector $X$ in Compressed-Vector Mode, and Zero the Same Specified Elements of $Y$</td>
<td>SGTHRZ</td>
<td>DGTHRZ</td>
</tr>
<tr>
<td>Multiply a Sparse Vector $X$ in Compressed-Vector Storage Mode by a Scalar, Add to a Sparse Vector $Y$ in Full-Vector Storage Mode, and Store in the Vector $Y$</td>
<td>SAXPYI</td>
<td>DAXPYI</td>
</tr>
<tr>
<td>Dot Product of a Sparse Vector $X$ in Compressed-Vector Storage Mode and a Sparse Vector $Y$ in Full-Vector Storage Mode</td>
<td>SDDOT</td>
<td>ZDDOT</td>
</tr>
</tbody>
</table>

---

ESSL Products General Information
Matrix-Vector Linear Algebra Subprograms

Matrix-Vector Product for a General Matrix, Its Transpose, or Its Conjugate Transpose
- SGEMV*
- CGEMV*
- SGEMX^®
- SGEMTX^®
- DGEMV*
- ZGEMV*
- DGEMX^®
- DGEMTX^®

Rank-One Update of a General Matrix
- SGER*
- CGER*
- CGERC*
- DGER*
- ZGERU*
- ZGERC*

Matrix-Vector Product for a Real Symmetric or Complex Hermitian Matrix
- SSPMV*
- CHPMV*
- S SYMV*
- CHEMV*
- SSLMX^®
- DSPMV*
- ZHPMV*
- DSYMV*
- ZHEMV*
- DSLMX^®

Rank-One Update of a Real Symmetric or Complex Hermitian Matrix
- SSPR*
- CHPR*
- SSYR*
- CHER*
- SSLR1^®
- DSPR*
- ZHPR*
- DSYR*
- ZHER*
- DSLR1^®

Rank-Two Update of a Real Symmetric or Complex Hermitian Matrix
- SSPR2*
- CHPR2*
- SSYR2*
- CHER2*
- SSLR2^®
- DSPR2*
- ZHPR2*
- DSYR2*
- ZHER2*
- DSLR2^®

Matrix-Vector Product for a General Band Matrix, Its Transpose, or Its Conjugate Transpose
- SGBMV*
- CGBMV*
- SGBMV*
- ZGBMV*

Matrix-Vector Product for a Real Symmetric or Complex Hermitian Band Matrix
- SSBMV*
- CHBMV*
- SSBMV*
- ZHBVM*

Matrix-Vector Product for a Triangular Matrix, Its Transpose, or Its Conjugate Transpose
- STRMV*
- CTRMV*
- STPMV*
- CTPMV*
- STBMV*
- DTRMV*
- ZTRMV*
- DTPMV*
- ZTPMV*
- CTBMV*
- ZTB MV*

Matrix-Vector Product for a Triangular Band Matrix, Its Transpose, or Its Conjugate Transpose

Sparse Matrix-Vector Linear Algebra Subprograms

Matrix-Vector Product for a Sparse Matrix in Compressed-Matrix Storage Mode
- DSMMX

Transpose a Sparse Matrix in Compressed-Matrix Storage Mode
- DSMTM

Matrix-Vector Product for a Sparse Matrix or Its Transpose in Compressed-Diagonal Storage Mode
- DSDMX

Matrix Operations

Matrix Addition for General Matrices or Their Transposes
- SGEADD
- CGEADD
- DGEADD
- ZGEADD

Matrix Subtraction for General Matrices or Their Transposes
- SGESUB
- CGESUB
- DGESUB
- ZGESUB

Matrix Multiplication for General Matrices, Their Transposes, or Conjugate Transposes
- SGEMUL
- CGEMUL
- DGEMUL
- ZGEMUL
- DGEMLP^®
- ZGEMMS

Matrix Multiplication for General Matrices, Their Transposes, or Conjugate Transposes Using Winograd’s Variation of Strassen’s Algorithm
- SGEMMS
- CGEMMS
- DGEMMS
- ZGEMMS
## Linear Algebraic Equations

### Dense Linear Algebraic Equations

<table>
<thead>
<tr>
<th>Description</th>
<th>SGEF</th>
<th>DGEF</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Matrix Factorization</td>
<td>SGEF</td>
<td>DGEF</td>
</tr>
<tr>
<td></td>
<td>CGEF</td>
<td>ZGEF</td>
</tr>
<tr>
<td></td>
<td>SGETRF⁺</td>
<td>DGETRF⁺</td>
</tr>
<tr>
<td></td>
<td>CGETRF⁺</td>
<td>ZGETRF⁺</td>
</tr>
<tr>
<td></td>
<td>SGETRF§</td>
<td>DGETRF§</td>
</tr>
<tr>
<td></td>
<td>CGETRF§</td>
<td>ZGETRF§</td>
</tr>
<tr>
<td></td>
<td>SGES</td>
<td>DGES</td>
</tr>
<tr>
<td></td>
<td>CGES</td>
<td>ZGES</td>
</tr>
<tr>
<td>General Matrix, Its Transpose, or Its Conjugate Transpose</td>
<td>SGES</td>
<td>DGES</td>
</tr>
<tr>
<td></td>
<td>CGES</td>
<td>ZGES</td>
</tr>
<tr>
<td></td>
<td>SGESM</td>
<td>DGESM</td>
</tr>
<tr>
<td></td>
<td>CGESM</td>
<td>ZGESM</td>
</tr>
<tr>
<td></td>
<td>SGETRS⁺</td>
<td>DGETRS⁺</td>
</tr>
<tr>
<td></td>
<td>CGETRS⁺</td>
<td>ZGETRS⁺</td>
</tr>
<tr>
<td></td>
<td>SGETRS§</td>
<td>DGETRS§</td>
</tr>
<tr>
<td></td>
<td>CGETRS§</td>
<td>ZGETRS§</td>
</tr>
<tr>
<td>General Matrix Factorization, Condition Number</td>
<td>SGEFCD</td>
<td>DGEFCD</td>
</tr>
<tr>
<td>Reciprocal, and Determinant</td>
<td>SGEICD</td>
<td>DGEICD</td>
</tr>
<tr>
<td></td>
<td>SGETRI⁺</td>
<td>DGETRI⁺</td>
</tr>
<tr>
<td></td>
<td>DGETRI⁺</td>
<td>ZGETRI⁺</td>
</tr>
<tr>
<td>Positive Definite Real Symmetric or Complex Hermitian Matrix Factorization</td>
<td>SPPF</td>
<td>DPPF</td>
</tr>
<tr>
<td></td>
<td>SPOF</td>
<td>DPOF</td>
</tr>
<tr>
<td></td>
<td>CPOF</td>
<td>ZPOF</td>
</tr>
<tr>
<td></td>
<td>SPOTRF⁺</td>
<td>DPOTRF⁺</td>
</tr>
<tr>
<td></td>
<td>CPOTRF⁺</td>
<td>ZPOTRF⁺</td>
</tr>
<tr>
<td></td>
<td>SPOTRF§</td>
<td>DPOTRF§</td>
</tr>
<tr>
<td></td>
<td>CPOTRF§</td>
<td>ZPOTRF§</td>
</tr>
<tr>
<td>Positive Definite Real Symmetric Matrix Solve</td>
<td>SPPS</td>
<td>DPPS</td>
</tr>
<tr>
<td>Positive Definite Real Symmetric or Complex Hermitian Matrix Multiple Right-Hand Side Solve</td>
<td>SPOSM</td>
<td>DPOSM</td>
</tr>
<tr>
<td></td>
<td>CPOSM</td>
<td>ZPOSM</td>
</tr>
<tr>
<td></td>
<td>SPOTRS⁺</td>
<td>DPOTRS⁺</td>
</tr>
<tr>
<td></td>
<td>CPOTRS⁺</td>
<td>ZPOTRS⁺</td>
</tr>
<tr>
<td></td>
<td>SPOTRS§</td>
<td>DPOTRS§</td>
</tr>
<tr>
<td></td>
<td>CPOTRS§</td>
<td>ZPOTRSM</td>
</tr>
<tr>
<td>Positive Definite Real Symmetric Matrix Factorization, Condition Number Reciprocal, and Determinant</td>
<td>SPPFC</td>
<td>DPPFC</td>
</tr>
<tr>
<td></td>
<td>SPOFC</td>
<td>DPOFC</td>
</tr>
<tr>
<td></td>
<td>SPOFC</td>
<td>DPOFC</td>
</tr>
<tr>
<td>Symmetric Indefinite Matrix Factorization and Multiple Right-Hand Side Solve</td>
<td>DBSSV</td>
<td></td>
</tr>
<tr>
<td>Symmetric Indefinite Matrix Factorization</td>
<td>DBSTRF</td>
<td></td>
</tr>
<tr>
<td>Symmetric Indefinite Matrix Multiple Right-Hand Side Solve</td>
<td>DBSTRS</td>
<td></td>
</tr>
<tr>
<td>General Matrix Inverse, Condition Number Reciprocal, and Determinant</td>
<td>SGEICD</td>
<td>DGEICD</td>
</tr>
<tr>
<td></td>
<td>SGETRI⁺</td>
<td>DGETRI⁺</td>
</tr>
</tbody>
</table>
Positive Definite Real Symmetric Matrix Inverse, Condition Number Reciprocal, and Determinant

Solution of a Triangular System of Equations with a Single Right-Hand Side

Solution of Triangular Systems of Equations with Multiple Right-Hand Sides

Triangular Matrix Inverse

Banded Linear Algebraic Equations

General Band Matrix Factorization
General Band Matrix Solve
Positive Definite Symmetric Band Matrix Factorization
Positive Definite Symmetric Band Matrix Solve
General Tridiagonal Matrix Factorization
General Tridiagonal Matrix Solve
General Tridiagonal Matrix Combined Factorization and Solve with No Pivoting
General Tridiagonal Matrix Factorization with No Pivoting

Positive Definite Symmetric Tridiagonal Matrix Factorization
Positive Definite Symmetric Tridiagonal Matrix Solve
Triangular Band Equation Solve

Sparse Linear Algebraic Equations

General Sparse Matrix Factorization Using Storage by Indices, Rows, or Columns
General Sparse Matrix or Its Transpose Solve Using Storage by Indices, Rows, or Columns
General Sparse Matrix or Its Transpose Factorization, Determinant, and Solve Using Skyline Storage Mode
Symmetric Sparse Matrix Factorization, Determinant, and Solve Using Skyline Storage Mode
Iterative Linear System Solver for a General or Symmetric Sparse Matrix Stored by Rows
Sparse Positive Definite or Negative Definite Symmetric Matrix Iterative Solve Using Compressed-Matrix Storage Mode
Sparse Positive Definite or Negative Definite Symmetric Matrix Iterative Solve Using Compressed-Diagonal Storage Mode
General Sparse Matrix Iterative Solve Using Compressed-Matrix Storage Mode

Chapter 5. ESSL for AIX Subroutines

31
General Sparse Matrix Iterative Solve Using Compressed-Diagonal Storage Mode

**Linear Least Squares**

- Singular Value Decomposition for a General Matrix: SGESVF, DGESVF
- Linear Least Squares Solutions for a General Matrix Using the Singular Value Decomposition: SGESVS, DGESVS
- General Matrix QR Factorization: DGEQRF
- Linear Least Squares Solution for a General Matrix: DGELS
- Linear Least Squares Solution for a General Matrix with Column Pivoting: SGELLS, DGELLS

**Eigensystems Analysis**

- Eigenvalues and, Optionally, All or Selected Eigenvectors of a General Matrix: SGEEV, DGEEV
- Eigenvalues and, Optionally, the Eigenvectors of a Real Symmetric Matrix or a Complex Hermitian Matrix: SSPEV, DSPEV, CHPEV, ZHPEV
- Extreme Eigenvalues and, Optionally, the Eigenvectors of a Real Symmetric Matrix or a Complex Hermitian Matrix: SSPSV, DSSPV, ZHPSV
- Eigenvalues and, Optionally, the Eigenvectors of a Generalized Real Eigensystem, $Az = wBz$, where $A$ and $B$ are Real General Matrices: SGEGV, DGEGV
- Eigenvalues and, Optionally, the Eigenvectors of a Generalized Real Symmetric Eigensystem, $Az = wBz$, where $A$ is Real Symmetric and $B$ is Real Symmetric Positive Definite: SSYGV, DSYGV

**Fourier Transforms, Convolutions and Correlations, and Related Computations**

**Fourier Transforms**

- Complex Fourier Transform: SCFT, SCFT5
- Real-to-Complex Fourier Transform: SRCFT, DRCFT
- Complex-to-Real Fourier Transform: SCRFT, DCRFT
- Cosine Transform: SCOSF, DCOSF
- Sine Transform: SSINF, DSINF
- Complex Fourier Transform in Two Dimensions: SCFT2, DCFT2
- Real-to-Complex Fourier Transform in Two Dimensions: SRCFT2, DRCFT2
- Complex-to-Real Fourier Transform in Two Dimensions: SCRFT2, DCRFT2
- Complex Fourier Transform in Three Dimensions: SCFT3, DCFT3
- Real-to-Complex Fourier Transform in Three Dimensions: SRCFT3, DRCFT3
- Complex-to-Real Fourier Transform in Three Dimensions: SCRFT3, DCRFT3
### Convolutions and Correlations

<table>
<thead>
<tr>
<th>Description</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convolution or Correlation of One Sequence with One or More Sequences</td>
<td>SCON</td>
</tr>
<tr>
<td>Convolution or Correlation of One Sequence with Another Sequence Using a Direct Method</td>
<td>SCOND</td>
</tr>
<tr>
<td>Convolution or Correlation of One Sequence with One or More Sequences Using the Mixed-Radix Fourier Method</td>
<td>SCONF</td>
</tr>
<tr>
<td>Convolution or Correlation with Decimated Output Using a Direct Method</td>
<td>SDCON</td>
</tr>
<tr>
<td>Autocorrelation of One or More Sequences</td>
<td>SACOR</td>
</tr>
<tr>
<td>Mixed-Radix Fourier Method</td>
<td>SCORF</td>
</tr>
<tr>
<td>Direct Method</td>
<td>DDCON</td>
</tr>
<tr>
<td>Method</td>
<td>DDCOR</td>
</tr>
</tbody>
</table>

### Related Computations

<table>
<thead>
<tr>
<th>Description</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polynomial Evaluation</td>
<td>SPOLY</td>
</tr>
<tr>
<td>1-th Zero Crossing</td>
<td>SIZC</td>
</tr>
<tr>
<td>Time-Varying Recursive Filter</td>
<td>STREC</td>
</tr>
<tr>
<td>Quadratic Interpolation</td>
<td>SQINT</td>
</tr>
<tr>
<td>Wiener-Levinson Filter Coefficients</td>
<td>SWLEV</td>
</tr>
</tbody>
</table>

### Sorting and Searching

<table>
<thead>
<tr>
<th>Description</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sort the Elements of a Sequence</td>
<td>SSORT</td>
</tr>
<tr>
<td>Sort the Elements of a Sequence and Note the Original Element Positions</td>
<td>SSORTX</td>
</tr>
<tr>
<td>Sort the Element of a Sequence Using a Stable Sort and Note the Original Element Positions</td>
<td>SSORTS</td>
</tr>
<tr>
<td>Binary Search for Elements of a Sequence X in a Sorted Sequence Y</td>
<td>SBSRCH</td>
</tr>
<tr>
<td>Sequential Search for Elements of a Sequence X in the Sequence Y</td>
<td>SSSRCH</td>
</tr>
</tbody>
</table>

### Interpolation

<table>
<thead>
<tr>
<th>Description</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polynomial Interpolation</td>
<td>SPINT</td>
</tr>
<tr>
<td>Local Polynomial Interpolation</td>
<td>STPINT</td>
</tr>
<tr>
<td>Cubic Spline Interpolation</td>
<td>SCSINT</td>
</tr>
<tr>
<td>Two-Dimensional Cubic Spline Interpolation</td>
<td>SCSIN2</td>
</tr>
</tbody>
</table>

### Numerical Quadrature

<table>
<thead>
<tr>
<th>Description</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numerical Quadrature Performed on a Set of Points</td>
<td>SPTNQ</td>
</tr>
<tr>
<td>Numerical Quadrature Performed on a Function Using Gauss-Legendre Quadrature</td>
<td>SGLNQ</td>
</tr>
<tr>
<td>Numerical Quadrature Performed on a Function Over a Rectangle Using Two-Dimensional Gauss-Legendre Quadrature</td>
<td>SGLNQ2</td>
</tr>
</tbody>
</table>

### Chapter 5. ESSL for AIX Subroutines
Numerical Quadrature Performed on a Function Using Gauss-Laguerre Quadrature
Numerical Quadrature Performed on a Function Using Gauss-Rational Quadrature
Numerical Quadrature Performed on a Function Using Gauss-Hermite Quadrature

**Random Number Generation**

- Generate a Vector of Uniformly Distributed Random Numbers
- Generate a Vector of Normally Distributed Random Numbers
- Generate a Vector of Long Period Uniformly Distributed Random Numbers

**Utilities**

- ESSL Error Information-Handler Routine
- ESSL ERRSAV Subroutine for the Workstations
- ESSL ERRSET Subroutine for the Workstations
- ESSL ERRSTR Subroutine for the Workstations
- Set the Vector Section Size (VSS) for the ESSL/370 Scalar Library
- Set the Extended Vector Operations Indicator for the ESSL/370 Scalar Library
- Determine the Level of ESSL Installed
- Determine the Stride Value for Optimal Performance in Specified Fourier Transform Subroutines
- Convert a Sparse Matrix from Storage-by-Rows to Compressed-Matrix Storage Mode
- For a General Sparse Matrix, Convert Between Diagonal-Out and Profile-In Skyline Storage Mode
- For a Symmetric Sparse Matrix, Convert Between Diagonal-Out and Profile-In Skyline Storage Mode

- DGRAD
- DHMQG
- SURAND
- DNRAND
- SURXOR

- EINFO
- ERRSAV
- ERRSET
- ERRSTR
- IVSSET
- IEVOPS
- IESSL
- STRIDE
- DSRSM
- DGKTRN
- DSKTRN
Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user’s responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.
IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Department LJEB/P905
2455 South Road
Poughkeepsie, NY 12601-5400
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

**Trademarks**

The following terms are trademarks of the International Business Machines Corporation in the United States or other countries or both:

AIX
AIX 5L
e (logo)
IBM
IBMLink
PowerPC
pSeries
RS/6000
Software Update Protocol

IBM has provided modifications to this software. The resulting software is provided to you on an “AS IS” basis and WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESS OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
Program Number: 5765-C41, 5765-C42