

***“The opportunity to work with such code still excites me greatly.***

***The potential is awesome, and I expect to begin producing something wonderful over the next few months.”***

***Nigel Nunn  
AIS, Australia***



**inuTech GmbH  
Fürther Str. 212  
90429 Nürnberg  
Germany**

**Phn: +49 911 32 38 43 - 0  
Fax: +49 911 32 38 43 - 43  
Eml: [diffpack@inutech.de](mailto:diffpack@inutech.de)**

**[www.diffpack.com](http://www.diffpack.com)  
[www.inutech.de](http://www.inutech.de)**

## **Wide Application Scope**

Diffpack is an object-oriented problem-solving environment for the numerical solution of partial differential equations (PDEs). Customer applications cover a wide range of engineering and scientific areas and span from simple educational applications to projects involving several man-years of development.

Customers such as Intel, Petrobras, DaimlerChrysler, NASA, Canon, CEA, Shell, Siemens, Mitsubishi, Sandia Nat. Lab., US Air Force, Natexis BP and universities such as Stanford, Cambridge, Harvard, Cornell, Rice, Chalmers and EPFL are using Diffpack in areas like geology, geophysics, chemistry, finance, medicine, physics, electronics, mechanics and mathematics.

## **How to get Started**

You are required to have a minimum background in the mathematics of PDEs as well as experience with programming in general. As the Diffpack learning process is supported by a comprehensive volume published by Springer-Verlag, only minor knowledge of C++ is necessary to get started.

## **Comprehensive PDE Functionality**

Diffpack is organized as a collection of C++ libraries embedded in an environment of software engineering tools.

The Diffpack Kernel contains a substantial collection of data structures and algorithms such as, e.g. vectors, matrices, strings, I/O, databases, reporting tools, GUIs, pre- and postprocessing tools, linear and non-linear systems and solvers, finite element and finite difference algorithms, grids, fields, numerical integration, real and complex arithmetic, etc. Functionality including e.g. adaptive meshing, multigrid methods and parallel computing is provided in separate toolboxes.

A major advantage is the high interoperability of these entities, giving you increased ability to construct and maintain complicated applications.

## **From Basic Models to Advanced Coupled Problems**

Typically, you will concentrate almost entirely on the essential and critical numerics of your problem. Tasks that are related to general numerics and program management are automatically handled by built-in Diffpack functionality.

You can build a basic FEM solver by coding a few functions, together small enough to fit on one or two sheets of paper. Still it will provide run-time access to a wide selection of numerical solution methods and reporting mechanisms.

Advanced compound simulators, solving systems of coupled PDEs, can be constructed by linking simpler sub-simulators together using well-defined principles from object-oriented programming realized in Diffpack.

(turn page)



***“I suspect there is little you cannot do with this package in the field of numerical modeling ...***

***it may be the first opportunity to solve many of the problems engineers have cheated on for years.”***

***Jim Moore  
Linux Journal***

***“I have been using Diffpack in my research for over one year and it has helped me immensely.”***

***Dave Shuster  
University of Virginia  
USA***

### **Plug'n Play at Run-Time**

Diffpack's menu system allows run-time selection of all application entities, from simple numerical parameters to abstract quantities such as element types, matrix formats, etc. You can use the menu system to set up advanced numerical experiments, e.g. looping over different solvers or preconditioners.

### **Extending Diffpack and Interfacing Your Legacy Code**

You can register your own entities in Diffpack's entity tables to make them fully interoperable with Diffpack. Existing code, in e.g. FORTRAN, can also be made interoperable via a thin communication interface. You can thus extend Diffpack into a tool tailored to your particular area of interest.

### **Pre- and Postprocessing**

The Diffpack Kernel provides preprocessors that support lattice grids, transfinite mappings and super-element techniques as well as interfaces to GeomPack and Triangle. Preprocessing is further extended by the Diffpack Datafilter Toolbox. Postprocessing supports tools such as Matlab, Gnuplot, IRIS Explorer, AVS and Vtk.

### **Documenting Your Experiments**

Full project reports containing problem description, source code, numerical results, images and movies can be generated in ASCII, LaTeX or HTML formats. The reporting tools interface the menu system, and you can use it to create reports of multiple runs with overview summaries and links to details.

### **Quality Assurance**

Diffpack guides you to a safe and consistent programming style, and gives you a variety of tools to enhance the quality of your development, including automatic memory handling, powerful debugging tools, regression testing and code transformation programs between Diffpack versions.

### **Flexibility and Efficiency**

In Diffpack, low-level computing intensive operations are always performed in a FORTRAN-like style, while object-oriented principles are only used for higher level administrative tasks. This ensures flexible APIs and computational efficiency competing with tailored FORTRAN codes.

### **From Prototype to Production Code**

Computing intensive applications are often developed by first utilizing general Diffpack tools to create a prototype. Next you localize bottlenecks and replace them with functions exploiting the structure of your problem. The replacements are made locally by overriding library functions using standard C++ constructs.