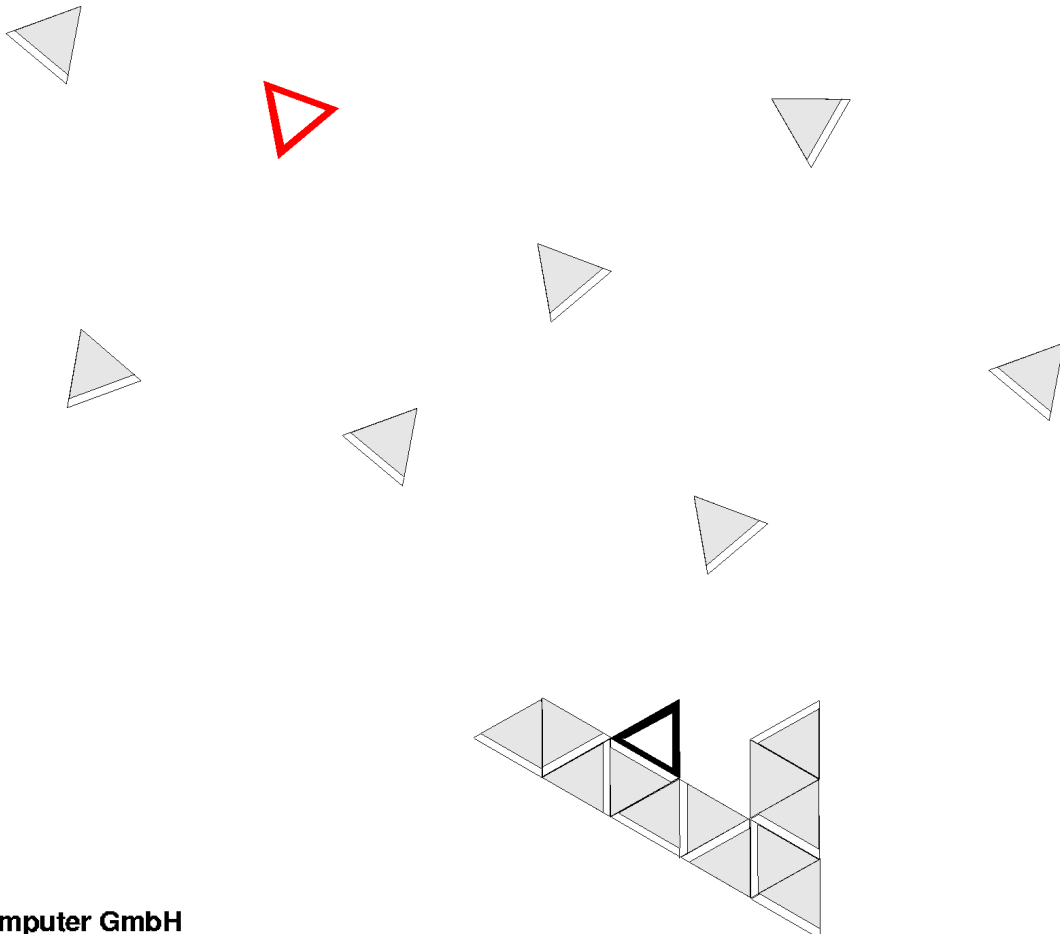


IF/Prolog



Product Line

Perfection in Prolog and Constraint Programming



IF/Prolog - a high-productivity, “declarative”, industry proven programming environment

IF/Prolog is one of the most well known and respected Prolog systems in use today. It has established its niche amongst other programming languages as the choice for special purpose use.

With its interfaces to graphics, Java, C/C++ and relational databases, IF/Prolog can be effectively embedded in a wide range of industrial applications.

Efficient data structures like binary trees and hash tables are generated automatically; search, pattern matching, and alternate solution finding are built into Prolog and provide powerful problem solving mechanisms.

As a result, programs are more concise, easier to maintain, and have shorter idea-to-market cycles.

Constraint Prolog - a powerful tool for resource management

Constraint programming offers a flexible, general purpose way to tackle almost any resource management task. Resource Management is the planning and allocation of different resources to perform tasks, ensuring their efficient utilisation and match maintenance and availability constraints.

Resources can be taken from any environment, for example: personnel, vehicles, aircraft, rooms, sports facilities, manufacturing machines, chemical processes, airport terminal gates, rail infrastructure...

Constraint Prolog is particularly well suited to finding efficient solutions to resource management taking into account both optimization and problem heuristics.

A good solution is one which is efficient and flexible enough to meet needs from personnel, marketing and implementation of the plan.

IF/Prolog for Windows – Windows for IF/Prolog

The new generation of IF/Prolog programmers use a comprehensive development environment compatible with Microsoft Developer Studio.

It is a multi-paradigm programming language environment which interacts with graphic objects generated using the graphical tools of developer studio.

IF/Prolog supports a multitude of Windows communication interfaces allowing programs to be constructed from several tools. Existing code may be easily integrated into the completed application.

IF/Prolog is ISO/IEC 13211/1 standard compliant and available on all MS-Windows platforms.

Application Areas:

- Intelligent Agents
- Design
- Decision Support
- Scheduling
- Resource Allocation
- Timetabling
- Expert Systems

Visual Prolog Development

A programming environment compatible with Microsoft Developer Studio allows rapid-development of all aspects of a Prolog application. The familiar studio environment handles the graphical editing of graphic objects and allows integration of IF/Prolog, Java, C/C++ and Visual BASIC code.

Microsoft Developer Studio templates are included to provide the starting point for any new application.

IF/Prolog may be embedded at any level, either taking full control of all Windows menus, events and dialog boxes or being used simply as a backend COM component.

IF/Prolog is compatible with both debug and release targets of developer studio and facilitates the debugging of Prolog code using the IF/Prolog development kernel.

A Multitude of Windows Interfaces

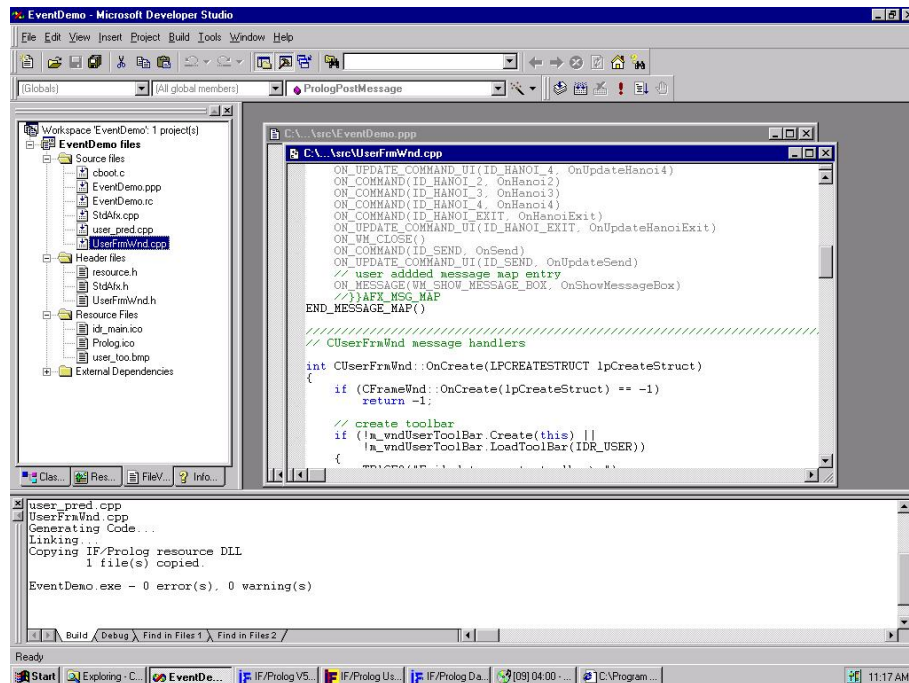
The COM- Interface provides a bi-directional communication to other tools such as Visual BASIC, Visual C++, Java, Oracle, Access, Excel...

The DDEML- Interface provides a bi-directional communication to other tools via DDE and DLLs.

The Event Interface provides an easy way to receive and send Windows events.

Socket Support Direct support for internet sockets and high level I/O streams enables Prolog to be embedded into intranet / internet applications.

Foreign Language Interfaces A flexible bi-directional interface to Java and C/C++ enables IF/Prolog programs to be embedded as a software component.



The Java Language Interfaces

A flexible **bi-directional** interface to Java enables IF/Prolog programs to be embedded as a software component for modern application development. IF/Prolog may either be called via a Java Class or may include Java implemented predicates.

A two level Java to Prolog interface – enables the programmer to either pass Prolog goals as strings which are then parsed, or to efficiently build up, in the systems internal representation, structures using a pre-defined set of macros.

The Prolog to Java interface – allows deterministic predicates to be defined as Java routines. These predicates can be as efficient as IF/Prologs own built-in predicates.

Java and Prolog together – Prolog, Java and C code may be freely inter-mixed and used both recursively and in re-entrant combinations. There are no size restrictions placed on data which can be exchanged between Java and Prolog as garbage collection and memory expansion may take place transparently, even during the execution of Java routines.

Java predicates in modules – Java routines may be assigned to modules which can then be incorporated as freely into large applications as modules written in Prolog.

The kernel of IF/Prolog:

At the kernel the IF/Prolog system is the ISO-Standard programming language Prolog. The standard ensures upwards compatibility and longevity of applications, IF/Prolog ensures performance and memory efficiency, integrating:

The Incremental Optimizing Compiler allowing incremental compilation, decompilation and loading of compiled code.

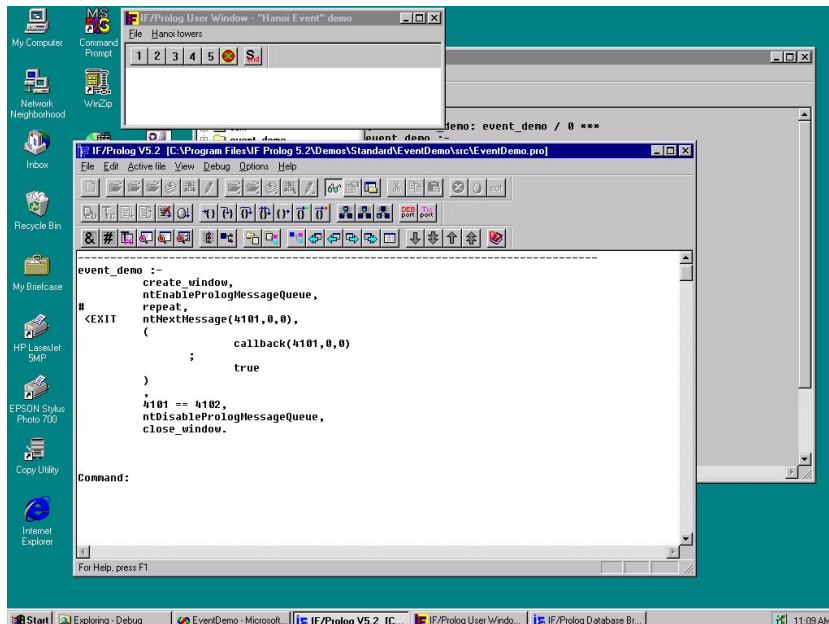
The Predicate Based Module Concept supporting data-hiding and meta-programming in a modular way.

Signal and Exception Handling to react to signals from the operating system and fully interact as a system process.

The Interactive Debugger to step through source code, both forward and backward. It has special features for constraint programming; showing suspension, activation and to examine constraint variables.

The Database Browser to list and examine predicate definitions in individual modules.

On-line Hypertext Help a comprehensive, indexed manual at your fingertips.



Constraint Technology

Constraint programming offers a flexible, general purpose way to tackle complex industrial optimization problems. It is in use in applications as diverse as production planning, crew rostering, 2D roll planning or school timetabling.

A constraint is a relation between variables which restricts the number of possible assignments each variable may take. Layers of constraints are used to reduce the problem in different ways, reducing the overall search space. Global constraints provide efficient constraint propagation between large groups of variables.

Powerful builtin solvers use well established Operations Research techniques to optimize but simultaneously obey the specified constraints and solve the problem efficiently. Different solvers and constraint routines can be combined incrementally with heuristic steps, implemented as Prolog rules. This combination of optimization and search leads more productively to a specific problem solution.

Constraint technology is an incremental development strategy which may be tailored to meet your customers requirements as they change.

IF/Prologs' constraints are of four types:

Finite Domain Constraints

specifies constraints over integers as ranges, intervals and sets.

Global Constraints

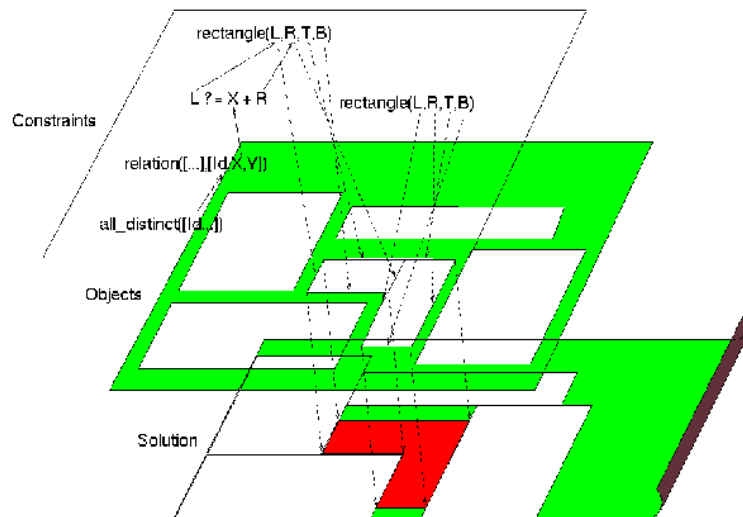
numerous builtin library constraint routines used to construct complex relations between many variables and enhance performance by utilising special internal propagation mechanisms.

Boolean Constraints

relations formulated with boolean expressions provide efficient decision tree manipulation.

Meta Constraints

allow new constraints to be applied dynamically based on the outcome of other constraints.



System Requirements

IF/Prolog is available on all major UNIX and MS-Windows platforms. Please note: additional software components will be required in conjunction with the interfaces.

| | memory | disk space |
|---------------------------------|--------|------------|
| IF/Prolog Development System | 64 MB | 50 MB |
| Constraint Technology Package | 64 MB | 20 MB |
| SQL Database (only UNIX) | 64 MB | 10 MB |
| OSF/Motif Interface (only UNIX) | 64 MB | 10 MB |

Customer Support

IF Computer supports its clients in all phases of a software project. Please refer to our applications brochure which presents many different projects as examples of successful development and cooperation with our clients.

We offer turn key solutions as well as partnerships in mixed project teams, to provide efficient know how transfer and if required long term support or further development work.

Company Profile

IF Computer is located in Germany and operates successfully in the world markets of knowledge engineering and complex optimisation. We utilise existing and newly emerging technologies to deliver client-server and intranet based solutions.

IF Computer provides quality project development, support, training and software consultancy; covering more native languages, continents and cultural backgrounds than any other specialist knowledge engineering company.

Synergy through sales of products, support and project work enables us to provide the maximum benefit of this technology to all of our customers.

IF Computer

www.ifcomputer.de
IF Computer GmbH
Cecinastr. 37
D-82205 Gilching
Tel: +49-8105-773550
Fax: +49-8105-773553
info@ifcomputer.de