



Constraint Logic Programming System

Overview

ECLiPSe is a software system - based on the CLP paradigm - for the development and deployment of constraint programming applications, for instance in the areas of planning, scheduling, resource allocation, timetabling and transportation.

ECLiPSe is also ideal for teaching most aspects of combinatorial problem solving, e.g. problem modelling, constraint programming, mathematical programming, and search techniques. Its wide scope makes it a good tool for research into hybrid problem solving methods.

ECLiPSe comprises several constraint solver libraries, a high-level modelling and control language, interfaces to third-party solvers, an integrated development environment and interfaces for embedding into host environments.

Availability

ECLiPSe has been open-sourced by Cisco Systems, Inc. under the Mozilla Public Licence, allowing unrestricted use for any purpose. Sources and binary releases are held at www.sourceforge.net/projects/eclipse-clp and at www.eclipse-clp.org.

Core Language Features

ECLiPSe is largely backward-compatible with **Prolog** and supports different dialects.

It has an extended set of **basic data types**: byte strings, unlimited precision integer and rational numbers, single and double precision floats. **Array** syntax and **structures** with field names are supported and are especially useful in constraint modelling.

A logical **iteration** construct eliminates the need for most simple recursion patterns.

ECLiPSe provides comprehensive facilities to implement **data-driven control** behaviour. These include declarative delay-clauses as well as primitives for meta-programmed con-

trol like explicit goal suspension, flexible triggering facilities and execution priorities.

The **attributed variable** data type is the key to many extensions to the basic Logic Programming language. The system calls user-definable event handlers when it encounters attributed variables in certain contexts, e.g. unification.

The **module system** controls the visibility of predicates, non-logical stores, source transformations and syntax settings. Module interfaces can be extended and restricted, and modules written in different language dialects can be mixed within one application.

Programs may contain **structured comments** from which reference documentation can be generated.

Libraries

ECLiPSe provides several libraries of **constraint solvers** which can be used in application programs: Arithmetic constraints over finite domains, finite set constraints (Conjunto), generalized propagation (Propia), interval reasoning over non-linear constraints, interfaces to external simplex solvers, constraint handling rules (CHR) and more.

Other libraries implement **search methods** like branch-and-bound, repair-based search, limited discrepancy search.

ECLiPSe interfaces to **external solvers**, in particular the COIN-OR, CPLEX[®] and Xpress-MP linear and mixed-integer programming solvers.

To simplify porting tasks, **compatibility libraries** for ISO Standard Prolog and other Prolog dialects are provided.

Various other **utility libraries**, including a number of popular public-domain ones, are included in the distribution.

Engineering

ECLiPSe is in commercial use and is robust and efficient. It is being actively maintained, and work is ongoing to boost performance as well as to develop new functionality.

The system is designed to impose no unnecessary limits on programs (number of variables, clause size) and data (size of strings, atoms, integers, structures). Stacks and dictionary are garbage collected.

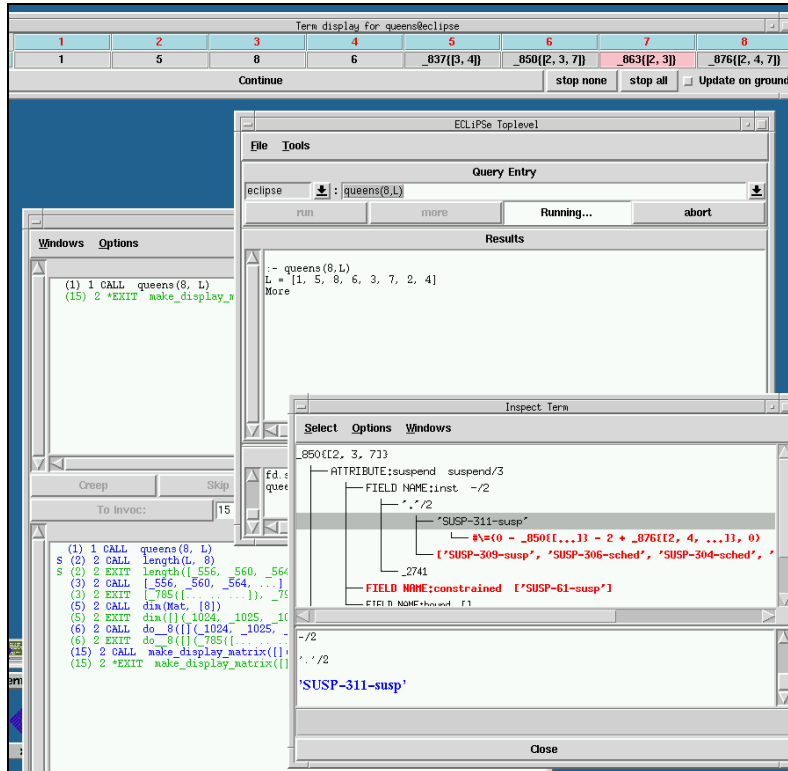
Embedding and Interfaces

ECLiPSe code can call C/C++ code, or a C/C++ main program can call ECLiPSe as a library. Data can be converted between C and ECLiPSe representations, or alternatively, ECLiPSe data can be referenced from

C/C++ and C/C++ data can be referenced from ECLiPSe.

There is a scripting language interface for embedding into Tcl/Tk and Java host applications. The main characteristics of these interfaces are bi-directional event-driven communication and the ability to communicate complex data structures through a language-independent data representation.

Development Environment



ECLiPSe comes with a cross-platform (Unix/Windows/OSX) graphical development environment. An ECLiPSe/eclipse integration is underway under the project name Saros.

Literature

Detailed documentation in both printable form and html format is included with the distribution, and also accessible on the web site www.eclipse-clp.org. This includes a tutorial, as well as user and reference manuals.

The book *Constraint Logic Programming using ECLiPSe* by Krzysztof Apt and Mark Wallace, Cambridge University Press, 2006, is a practical introduction to constraint programming and to ECLiPSe, with many examples, appropriate for a one-semester course.

The book *Programming with Constraints: an Introduction* by Kim Marriott and Peter Stuckey, MIT Press, 1998, is a comprehensive introduction to constraint programming, with examples in ECLiPSe.