CP-INSIDE: Embedding Constraint-based Decision Engines in Business Applications

Jacob Feldman and Eugene Freuder and James Little*

Cork Constraint Computation Centre
Department of Computer Science
University College Cork
Cork, Ireland
{j.feldman, e.freuder, j.little}@4c.ucc.ie

The CP-INSIDE project seeks to make constraint programming technology more accessible. In particular, it aims to facilitate the embedding of CP technology in business applications, and its integration with familiar software tools. A major objective of the CP-INSIDE project is to allow business developers to create constraint-based decision engines utilizing the power of CP technology in a vendor-neutral way. CP-INSIDE simplifies the development of specialized decision engines making them independent of the underlying generic CP solvers. There have already been several attempts to unify constraint programming languages (for instance, [1–3]) that directly or indirectly contributed to the same objective. CP-INSIDE is based on the following principles:

- Use main-stream programming languages like Java giving business application developers a simple CP API with no new languages to learn and with an easy access to already existing business objects and packages.
- Provide pre-built interfaces to commonly used software products such as MS Excel, Google Apps, and different rule engines. Application developers should be able to use familiar languages, tools, and a learn-by-example technique instead of becoming CP gurus.
- Not to develop another CP solver, but rather externalize commonly used features from the most popular CP solvers and provide a unified way to build adapters for different, already existing solvers.
- Allow CP researches to implement new algorithms/constraints in a unified way, test them with different solvers, and make them commonly available.

While we considered both Java and C# for CP-INSIDE, the initial release has been implemented in Java as a three-tier framework presented on Figure 1. The first tier "Business Interfaces" uses the second tier "Common CP API" to provide pre-built unified interfaces between different CP Solvers and popular tools such as MS Excel, Google Calendar, Facebook, MATLAB, or business rules management systems such as OpenRules. It also provides examples of how to build web interfaces for CP-based engines and to deploy them as web services. The Common CP API provides a Java interface for major CP concepts and methods that can be utilized by business application developers to define and solve their own constraint satisfaction problems.

* This work is supported by Enterprise Ireland under Grant CFTD/06/209 and by the Science Foundation Ireland under Grant No. 05/IN/I886.

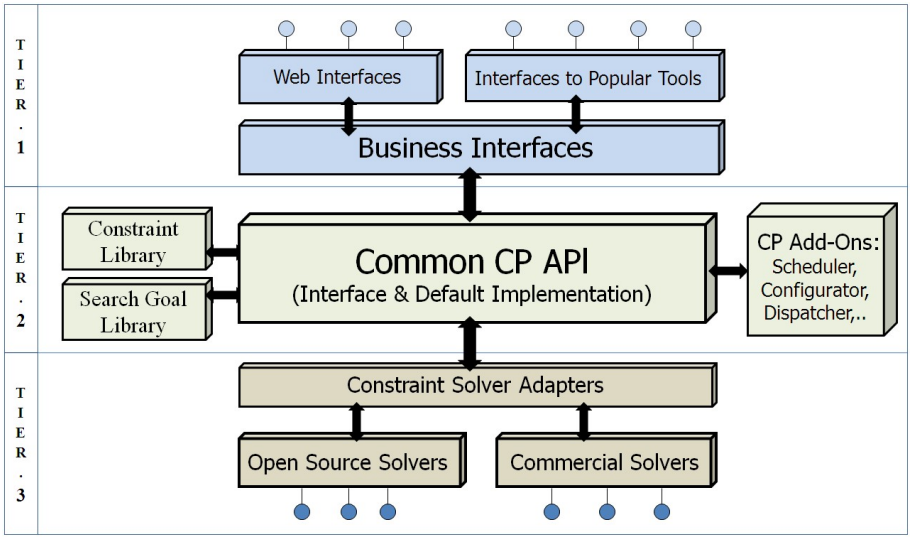

Fig. 1. CP-INSIDE Architecture

Along with an interface, the Common CP API includes default implementations for major constraints and search goals that do not depend on a specific underlying CP solver. However, the API also allows a user to utilize the power of a specific solver by providing an access to the native implementation. With the CP API a user may define new custom constraints and search goals without the necessity to go down to the CP solver level. CP-INSIDE includes a Scheduler built on top of the CP API, thus providing scheduling concepts, constraints, and goals even for solvers that do not support them. Similarly, we plan to build add-ons for other business verticals. The third tier provides a set of adapters for popular existing Java-based CP solvers. The initial set includes adapters for commercial ILOG JSolver (www.ilog.com) and open source CP solvers Choco (www.choco.sourceforge.net), Constrainer (www.constrainer.sourceforge.net). These adapters can be used as examples for integrating other Java-based CP solvers.

References

1. Van Hentenryck, P.: The OPL Optimization Programming Language. MIT Press, Cambridge, Mass (1999)
2. Schlenker, H., Ringwelski, G.: POOC - a platform for object-oriented constraint programming. In O'Sullivan, B., ed.: Recent Advances in Constraints: Joint ERCIM/CologNet International Workshop on Constraint Solving and Constraint Logic Programming, Springer Verlag (2002) 159–170
3. Nethercote, N., Stuckey, P., Becket, R., Brand, S., Duck, D., Tack, G.: MiniZinc: Towards a standard CP modelling language. In: Proceedings of the 13th International Conference on Principles and Practice of Constraint Programming (CP2007), Providence, RI (2007)