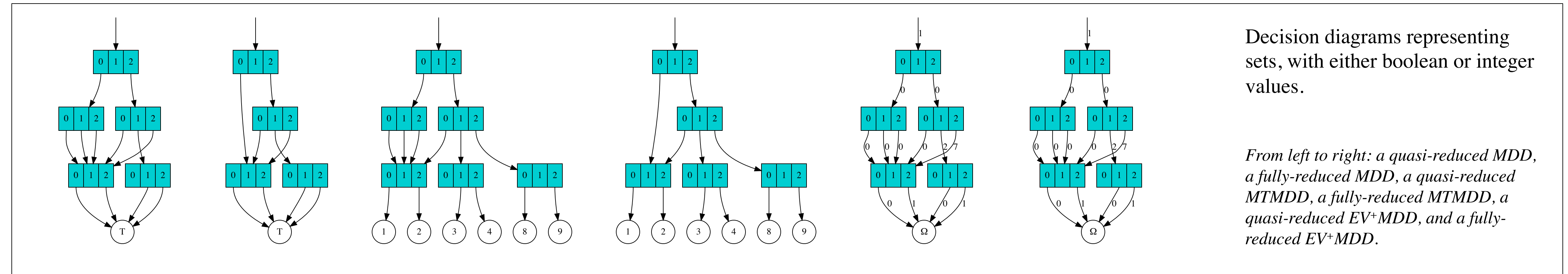


SI2-SSE: A next-generation decision diagram library

<http://meddly.sourceforge.net>

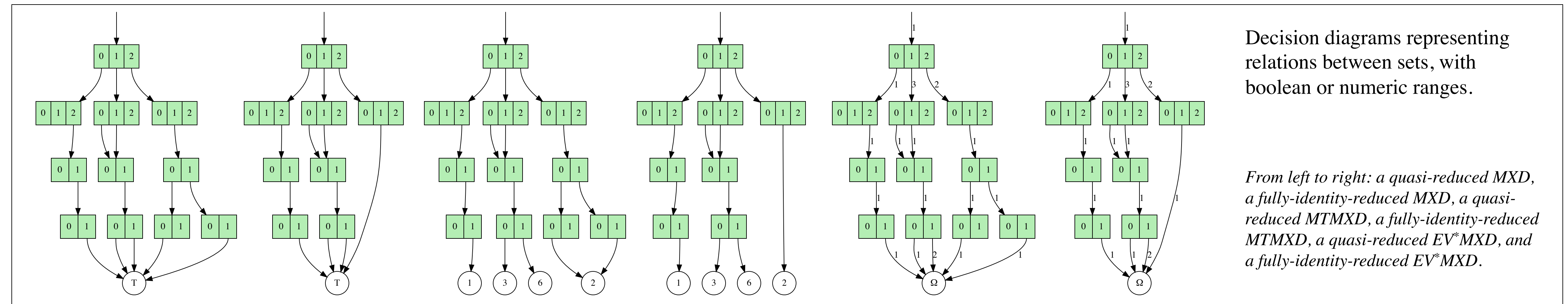
MEDDLY, Multi-terminal and Edge-valued Decision Diagram LibrarY, is an open-source decision diagram library written in C++. It has been built to natively support multi-terminal and edge-valued decision diagrams.

Overview. Many computer-based scientific or engineering applications need to store, analyze, and manipulate large data. Often, this data has enough structure that specialized data structures and algorithms can have dramatically smaller memory and time requirements than explicit approaches. An important such case is symbolic verification of hardware and software, where, traditionally, binary decision diagrams (BDDs) have been successfully employed to study systems with enormous state spaces. Several software libraries for BDDs have arisen to support these operations, and BDDs have been applied to diverse applications as a means to exploit structure that is often hidden.



Decision diagrams representing sets, with either boolean or integer values.

From left to right: a quasi-reduced MDD, a fully-reduced MDD, a quasi-reduced MTMDD, a fully-reduced MTMDD, a quasi-reduced EV+MDD, and a fully-reduced EV+MDD.



Decision diagrams representing relations between sets, with boolean or numeric ranges.

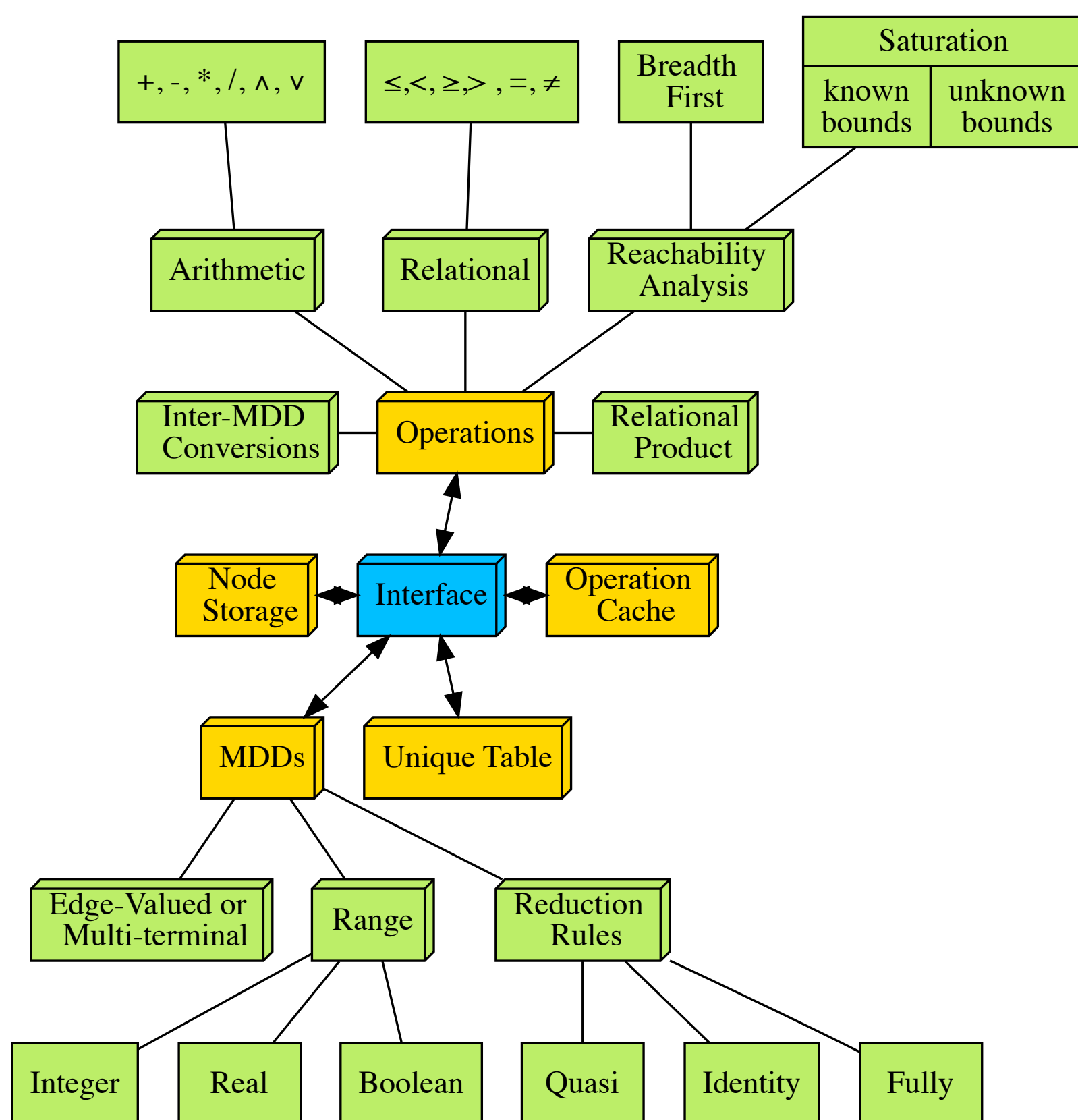
From left to right: a quasi-reduced MXD, a fully-identity-reduced MXD, a quasi-reduced MTMXD, a fully-identity-reduced MTMXD, a quasi-reduced EV*MXD, and a fully-identity-reduced EV*MXD.

However, in the past decade, decision diagram theory has continued to advance, by generalizing BDDs to variants with multi-way decisions (MDDs), multi-way or numerical outcomes, or edge values to encode real-valued data, and by proposing a variety of reduction rules to change (and often shrink) the decision diagram shape, as well as many important algorithmic improvements. Unfortunately, decision diagram libraries have not kept up with these theoretical advances.

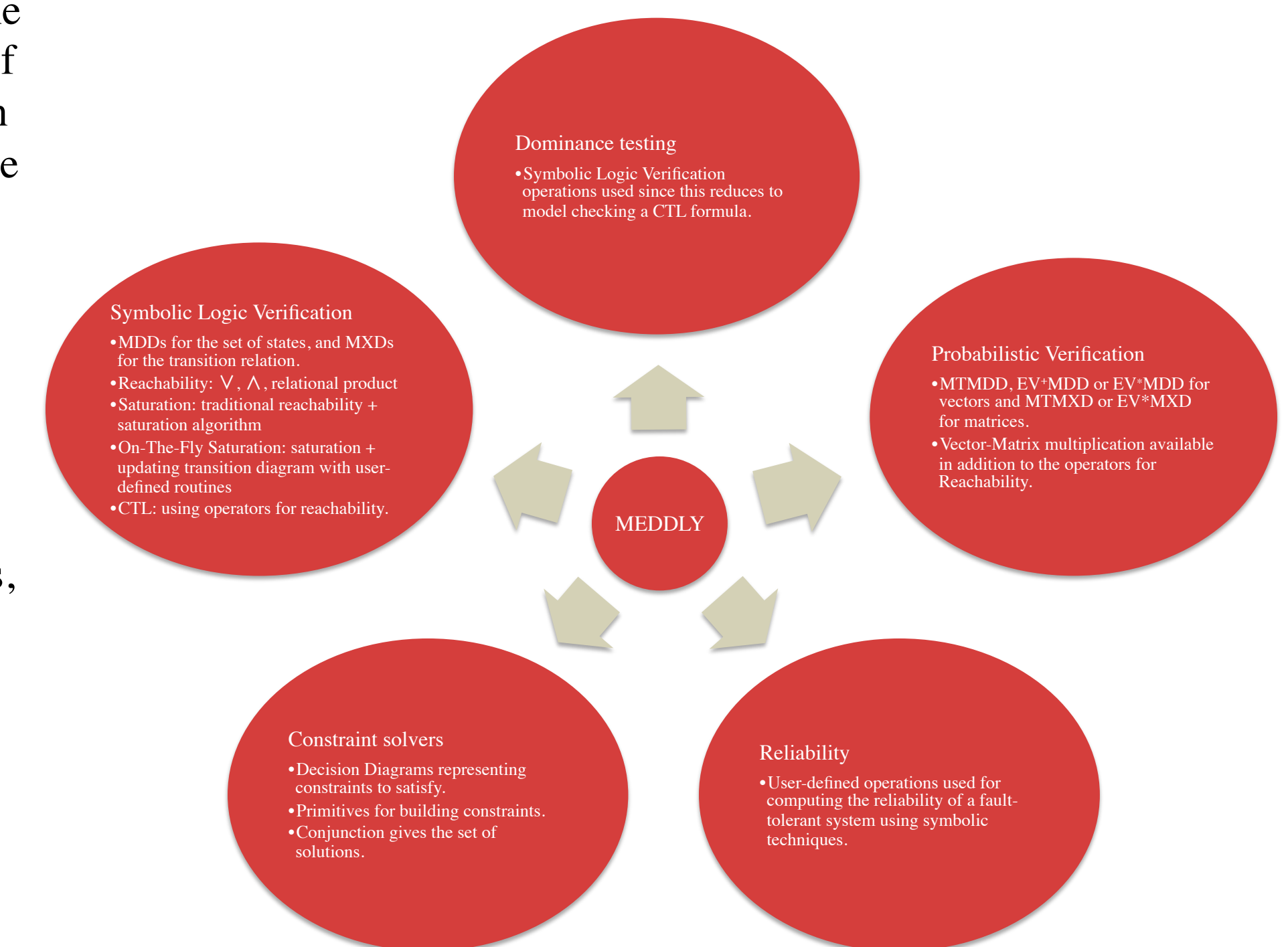
MEDDLY seeks to fill this gap, by merging and expanding two existing prototype libraries, the previous version of *MEDDLY* (designed by PI Miner) and *TEDDY* (designed by Co-PI Ciardo), into a powerful, next-generation decision diagram library that supports a more general theory of decision diagrams. The new library will encompass (1) non-binary variables, including a-priori unbounded discrete domains and even infinite domains under certain restrictions, (2) non-boolean function values,

attached either to terminal nodes or to the edges of the decision diagram, and (3) a more general definition of canonicity that includes a wide spectrum of reduction rules. Several proposed activities will help smooth the learning curve for users adopting this library, from proven methods such as user manuals, tutorials, examples, wikis, and user groups, to novel ones such as the development of visualization techniques to aid the debugging and understanding of decision diagrams.

Modularity has been one of *MEDDLY*'s design goals, and all its major components interact via the user-interface. This makes *MEDDLY* easy to extend, and (potentially) to parallelize. Users could define their own reduction rules, caching schemes and decision diagram operations without requiring extensive knowledge of *MEDDLY*'s internals.



System Diagram for *MEDDLY*. All major components interact using the user interface, maintaining modularity and extensibility.



Major applications for *MEDDLY*, and supported operations.