

Процик Петр Павлович,
студент кафедры теории и технологии программирования Киевского национального
университета имени Тараса Шевченко
Адрес – 03127, Киев, ул. Сеченова, 6 общ. 16, ком. 132,
E-mail: ppp_extr@univ.kiev.ua

О РЕАЛИЗАЦИИ СИСТЕМЫ САД В СРЕДЕ WINDOWS

Процик П.П.

Система Автоматизации Дедукции (САД) представляет собой реализацию идей, предложенных В. Глушковым в рамках программы работ по Алгоритму Очевидности [1,2]. Система позволяет производить обработку формальных математических текстов, написанных на формальном языке ForTheL [3]. Отличительной особенностью языка является его близость к языкам естественных математических публикаций и возможность его трансляции в язык исчисления предикатов первого порядка.

В текущий момент времени система САД реализована в среде операционных систем Unix/Linux и Windows. Она позволяет выполнять следующие действия:

- синтаксический анализ и трансляция в язык логики первого порядка ForTheL- текстов;
- производить поиск вывода в логиках первого порядка с использованием как внешних, так и собственных средств;
- поиск доказательства утверждений и теорем, содержащихся в ForTheL - тексте;
- верификация ForTheL- текстов.

К Linux-версии системы доступ можно получить через web-страницу проекта "<http://ea.unicyb.kiev.ua>", а описания такой ее реализации представлены в [4,5].

Технической особенностью системы САД является её модульная архитектура, т.е. она состоит из функционально независимых взаимодействующих единиц. Концептуальная структура системы представлена на схеме 1.

С целью способствовать её использованию конечным пользователем, имеющим навык работы в операционной системе Windows, было принято решение реализовать персональный Linux-вариант системы САД в среде Windows с сохранением всех функций оригинальной версии и предоставлением удобного интерфейса.

При реализации Windows версии, наибольшей переработке подверглась часть системы, отвечающая за взаимодействие с внешними подключаемыми модулями. Это вызвано, прежде всего, большой зависимостью внешних модулей от операционной системы, методов обмена сообщениями между потоками и способом организации взаимодействия процессов. На схеме, за поддержку такого взаимодействия в системе отвечают модули Explore и Naigha. Первый из них формулирует задания для модулей логического вывода (prover), а второй (в случае применения постороннего прувера) занимается трансляцией этого задания в формат входных данных для каждого конкретного prover'a, а также интерпретацию результатов его работы.

В рамках Windows-версии системы САД был реализован собственный прувер, использующий оригинальные процедуры поиска логического вывода. В качестве альтернативы можно использовать резолюционный прувер с обработкой равенства SPASS (как внешний модуль расширения). Кроме того, в тестовом режиме можно подключать Otter и Vampire.

Немаловажной частью работы являлась разработка удобного графического интерфейса. Он предоставляет доступ ко всем функциям системы, а также позволяет использовать подключаемые к системе модули.

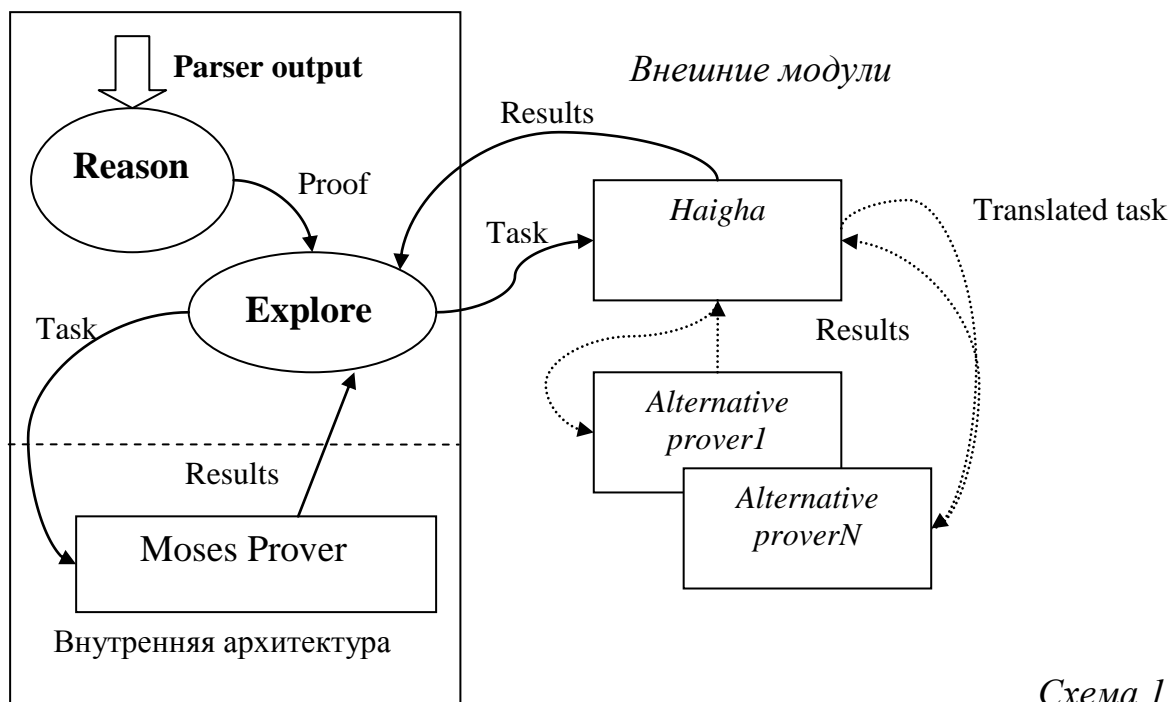


Схема 1

На данный момент ядро системы и все функциональные модули полностью перенесены под управление операционной системы Windows. Тестирование функциональности проводилось с использованием примеров из оригинальной версии, а также специально разработанных в ходе работ.

ЛИТЕРАТУРА:

1. Глушков В.М. Некоторые проблемы теории автоматов и искусственного интеллекта. Кибернетика, 1970, № 2, 3-13.
2. System for Automated Deduction, SAD (<http://ea.unicyb.kiev.ua/sad.en.html>)
3. Verchinine K. and Paskevich A.: ForTheL reference manual, Unpublished draft, hosted at <http://ea.unicyb.kiev.ua/download/forthel.ps.gz>, 2004
4. Alexander Lyaletski, Konstantin Verchinine, Anatoli Degtyarev, and Andrey Paskevich. System of Automated Deduction (SAD): Linguistic and Deductive Peculiarities. In M.A. Klopotek, S.T. Wierzchon, and M. Michaliwicz, editors, *Advances in Soft Computing: Intelligent Information Systems 2002*, pages Physica-Verlag, Springer, pages 413-422, 2002.
5. Lyaletski A., Verchinine K., and Paskevich A. On verification tools implemented in the System for Automated Deduction. *Proceedings of the 2nd Workshop on Implementation Technology for Computational Logic Systems (ITCLS)*, Pisa, Italy, 2003, 25—36