

Основы искусственного интеллекта



Языки логического программирования

Начало истории

В **октябре 1981** года Японское министерство международной торговли и промышленности объявило о создании исследовательской организации — Института по разработке методов создания компьютеров нового поколения (**Institute for New Generation Computer Technology Research Center**).

Целью данного проекта было создание систем обработки информации, базирующихся на знаниях.

Предполагалось, что эти системы будут обеспечивать простоту управления за счет возможности общения с пользователями при помощи естественного языка. Эти системы должны были

- самообучаться;
- использовать накапливаемые в памяти знания для решения различного рода задач;
- предоставлять пользователям экспертные консультации;
- от пользователя не требовалось быть специалистом в информатике.

Начало истории

Основная суть качественного перехода к пятому поколению ЭВМ заключалась в переходе от обработки данных к обработке знаний. Разработчики надеялись, что им удастся не подстраивать мышление человека под принципы функционирования компьютеров, а приблизить работу компьютера к тому, как мыслит человек, отойдя при этом от фон Неймановской архитектуры компьютеров.

В **1991 году** предполагалось создать первый прототип компьютеров пятого поколения.

Поставленные цели в полной мере так и не были достигнуты, однако проект послужил импульсом к развитию нового витка исследований в области искусственного интеллекта и вызвал взрыв интереса к логическому программированию. Так как для эффективной реализации традиционная фон Неймановская архитектура не подходила, были созданы специализированные компьютеры логического программирования PSI и PIM.

Языки логического программирования

В качестве основной методологии разработки программных средств для проекта **ЭВМ пятого поколения** было избрано **логическое программирование**, представителем которого является **язык Пролог**. В настоящее время Пролог остается наиболее популярным языком искусственного интеллекта в Японии и Европе (в США, традиционно, более распространен другой язык искусственного интеллекта — язык функционального программирования **Лисп**).

Название языка "**Пролог**" происходит от слов **ЛОГическое ПРО**граммирование (PROgrammation en LOGique во французском варианте и **PROgramming in LOGic** — в английском).

Пролог основывается на таком разделе **математической логики**, как **исчисление предикатов первого порядка**. Точнее, его базис составляет процедура доказательства теорем методом резолюции для хорновских дизъюнктов.

История возникновения и развития языка Пролог

В 1965 году в работе "A machine oriented logic based on the resolution principle", опубликованной в 12 номере журнала "Journal of the ACM", Дж. Робинсон представил метод автоматического поиска доказательства теорем в исчислении предикатов первого порядка, получивший название "принцип резолюции". Эту работу можно прочитать в переводе: Робинсон Дж. Машинно-ориентированная логика, основанная на принципе резолюции // Кибернетический сборник. — Вып. 7 (1970).

На самом деле, идея данного метода была предложена Эрбраном в 1931 году, когда еще не было компьютеров (Herbrand, "Une methode de demonstration", These, Paris, 1931). Робинсон модифицировал этот метод так, что он стал пригоден для автоматического, компьютерного использования, и, кроме того, разработал эффективный алгоритм унификации, составляющий базис его метода.

История возникновения и развития языка Пролог

В 1973 году "группа искусственного интеллекта" во главе с **Аланом Колмероз** создала в Марсельском университете программу, предназначенную для доказательства теорем. Эта программа использовалась при построении систем обработки текстов на естественном языке. Программа доказательства теорем получила название **Prolog** (от **Programmation en Logique**). Она и послужила прообразом Пролога. Программа была написана на Фортране и работала довольно медленно.

Большое значение для развития логического программирования имела **работа Роберта Ковальского "Логика предикатов как язык программирования"** (Kowalski R. Predicate Logic as Programming Language. IFIP Congress, 1974), в которой он показал, что для того чтобы добиться эффективности, нужно ограничиться использованием множества хорновских дизъюнктов. Известно, что Ковальский и Колмероз работали вместе в течение одного лета.

История возникновения и развития языка Пролог

В **1976 г.** Ковальский вместе с его коллегой Маартеном ван Эмденом предложил два подхода к прочтению текстов логических программ: процедурный и декларативный.

В **1977 году** в Эдинбурге **Уоррен и Перейра** создали очень эффективный **компилятор языка Пролог для ЭВМ DEC-10**, который послужил прототипом для многих последующих реализаций Пролога. Компилятор был написан на самом Прологе. Эта реализация Пролога, известная как "**эдинбургская версия**", фактически стала первым и единственным стандартом языка. Алгоритм, использованный при его реализации, послужил прототипом для многих последующих реализаций языка. Как правило, если современная Пролог-система и не поддерживает эдинбургский Пролог, то в ее состав входит подсистема, переводящая прологовскую программу в "эдинбургский" вид. Имеется **стандарт ISO/IEC 13211- 1:1995**, но его поддерживают далеко не все Прологсистемы.

История возникновения и развития языка Пролог

В 1980 году Кларк и Маккейб в Великобритании разработали версию Пролога для персональных ЭВМ.

В 1981 году стартовал вышеупомянутый проект Института по разработке методов создания компьютеров нового поколения.

На сегодня существует довольно много реализаций Пролога. Наиболее известные из них следующие: BinProlog, AMZI-Prolog, Arity Prolog, CProlog, Micro Prolog, МПролог, Prolog-2, Quintus Prolog, SICTUS Prolog, Silogic Knowledge Workbench, Strawberry Prolog, SWI Prolog, UNSW Prolog и т. д.

В России были разработаны такие версии Пролога как Пролог-Д (Сергей Григорьев), Акторный Пролог (Алексей Морозов), а также Флэнг (А. Манцивода, Вячеслав Петухин).

Реализации языка ПРОЛОГ

Исторически первым языком логического программирования был **Planner**, который так и не получил широкого распространения.

Каждая из реализаций ПРОЛОГ имеет свое назначение, особенности и возможности.

Это наиболее популярная, в свое время, реализация языка ПРОЛОГ. Использовалась как средство разработки и как учебное пособие в большинстве учебных заведений. Особенность данной реализации в том, что является компилятором языка, и позволяет достаточно просто реализовывать законченные решения. В данной реализации используется строгая типизация данных, что повышает скорость трансляции и выполнения программ.

Turbo Prolog

"Пролог" от Borland

Операционная система MS DOS

Реализации языка ПРОЛОГ



Visual Prolog

Это мощная система, включающая практически уникальную, среди реализаций языка ПРОЛОГ, графическую среду разработки. **Visual Prolog** - это развитие системы **Turbo Prolog** и наследует ее основные принципы, является достаточно хорошо документированной. Уникальная особенность системы в том, что система постоянно развивается и на ее базе разрабатываются и успешно работают множество коммерческих систем.

Достоинства:

- качественная среда визуальной разработки;
- объектно-ориентированный язык;
- строгая типизация данных;
- автоматическое управление памятью;
- поддержка C / C++;
- прямой вызов функций Win32 API.

Prolog Development Center

Операционная система: **Windows**

Домашняя страница:

<http://www.visual-prolog.com>

Реализации языка ПРОЛОГ



Ciao Prolog

Система **Ciao Prolog** является полной средой программирования с набором уникальных особенностей для разработки ПРОЛОГ программ. В основу данной реализации легла разработка диалекта **ISO Prolog**. **Ciao Prolog** является реализацией с открытым кодом (распространяется под лицензией **GNU GPL**).

Операционная система: **Windows, Linux**

Домашняя страница:

<http://www.clip.dia.fi.upm.es/Software/Ciao/>

Реализации языка ПРОЛОГ



SWI-Prolog



SWI Prolog

SWI-Prolog это открытая реализация языка программирования ПРОЛОГ. Данная реализация достаточно часто используется для обучения. В пакет включен большой набор функций и библиотек.

Особенности языка:

- программирование с ограничениями;
- многопоточность;
- возможность модульного тестирования;
- хороший графический интерфейс;
- IDE отладчик;
- интерфейсы для Java, ODBC и др.;
- качественная документация.

Операционная система: **Unix, Windows, Macintosh**
Домашняя страница: <http://www.swi-prolog.org/>

Реализации языка ПРОЛОГ



SICStus Prolog — это современная система разработки программ. В основе языка лежит стандарт ISO Prolog. **SICStus Prolog** достаточно эффективен и устойчив для большинства приложений.

Особенности:

- ❑ высокая эффективность;
- ❑ полная поддержка UNICODE;
- ❑ двухсторонний интерфейс к приложениям C, C++, .NET и Java;
- ❑ эффективный отладчик и обработчик событий;
- ❑ полная поддержка стандарта ISO/IEC 13211-1.

Операционная система:

Unix, Windows, Macintosh, Solaris

Домашняя страница:

<http://www.sics.se/isl/sicstuswww/site/>

Реализации языка ПРОЛОГ

Strawberry Prolog



Strawberry Prolog представляет собой очень простую в использовании реализацию языка программирования ПРОЛОГ. Его синтаксис очень близок к стандарту ISO-Prolog, но в нем также присутствуют и некоторые расширения.

Основная идея **Strawberry Prolog** - простота использования и поэтому данная реализация достаточно часто используется в университетах для изучения языка.

Операционная система:

Unix, Windows, FreeBSD, Solaris
Домашняя страница <http://www.dobrev.com>

Реализации языка ПРОЛОГ



Amzi Prolog



Amzi Prolog - достаточно мощная реализация языка ПРОЛОГ, постоянно расширяемая и поддерживаемая.

Основные функции:

- ❑ интеграция с другими языками;
- ❑ детальная документация.

Операционная система:

Unix, Windows, FreeBSD, Solaris

Домашняя страница:

<http://www.amzi.com>

История возникновения и развития языка Пролог

Наиболее известна в России версия Пролога — **Турбо Пролог**. Начиная разрабатывать фирма Borland International в содружестве с датской компанией Prolog Development Center (PDC). Первая версия вышла в **1986 году**. Последняя совместная версия имела номер 2.0 и была выпущена в **1988 году**.

В **1990 году** PDC получила монопольное право на Турбо Пролог и дальше продвигала его под названием **PDC Prolog**.

В **1992 году** вышла версия **PDC Prolog 3.31**.

В **1996 году**, при участии группы питерских программистов, **Prolog Development Center** выпустил систему **Visual Prolog 4.0**. В состав среды Visual Prolog были включены инструментальные средства генерации кода, конструирующие управляющую логику, интерфейс визуального программирования и другие средства, позволяющие ускорить разработку приложений. Помимо прочих достоинств **среды Visual Prolog** стоит обратить внимание на возможность использования в идентификаторах символов национального алфавита, в частности, можно употреблять в программах русские имена доменов, предикатов и переменных, что делает программу более понятной и самодокументированной.

Применения Пролог

Основные области применения Пролога:

- быстрая разработка прототипов прикладных программ;
- автоматический перевод с одного языка на другой;
- создание естественно-языковых интерфейсов для существующих систем;
- символьные вычисления для решения уравнений, дифференцирования и интегрирования;
- проектирование динамических реляционных баз данных;
- экспертные системы и оболочки экспертных систем;
- автоматизированное управление производственными процессами;
- автоматическое доказательство теорем;
- полуавтоматическое составление расписаний;
- системы автоматизированного проектирования;
- базирующееся на знаниях программное обеспечение;
- организация сервера данных или, точнее, сервера знаний, к которому может обращаться клиентское приложение, написанное на каком-либо языке программирования.

Области, для которых Пролог не предназначен: большой объем арифметических вычислений (обработка аудио, видео и т.д.); написание