

**Л. В. Городняя**

**ЭКСПЕРИМЕНТАЛЬНЫЕ СИСТЕМЫ  
ПРОГРАММИРОВАНИЯ ЛИТТЛ, ЛИСП**

Для многих учеников А. П. Ершова типичной темой дипломной работы была реализация какого-нибудь нового, модного, недоступного у нас или свежесочиняемого языка программирования. Некоторые разработки являлись чем-то вроде диалога с мировыми авторитетами в стиле интеллектуального вызова. Другие возникали как элементы технологии программирования в наших реальных условиях.

**Лисп-система на БЭСМ-6**

В конце 1968 года в кабинете А. П. Ершова Джон Маккарти прочел серию лекций, посвященных языку Лисп. Возможность практического применения такого языка даже на нашей лучшей ЭВМ БЭСМ-6 вызывала сомнения у специалистов.

**Преодоление препятствий**

Первая версия реализации Лисп-интерпретатора в Новосибирске была брошена после года работы из-за обнаруженного несовершенства исходных решений (хэш-таблица атомов оказалась непригодной к удалению объектов). Вторая версия экспериментальной реализации полного Lisp 1.5 в условиях получения пяти минут машинного времени в день, чего с трудом хватало на прогон одной колоды на автокоде, потребовала от Т. С. Янчук (вскоре мобилизованной в проект Альфа-6) изрядной изобретательности при выборе результативных проектных и технологических решений. Например, для представления интерпретатора она предложила специально сконструированный язык загрузчика, удачно сочетавший эффективность программирования в кодах с гибкостью символьной обработки. В результате за пять минут вполне удавалось пропустить более пяти колод по 1000 перфокарт. Протокол работы программы начинал печататься моментально, сразу вслед за завершением приема перфокарт. Это определило достаточную результативность отладки Лисп-интерпретатора.

## Возможности Лисп-системы

Производственная версия Лисп-системы включала в себя русифицированную лексику, универсальную обработку свойств объектов и механизм перераспределения памяти с выгрузкой стека во внешнюю память. Алгоритмы и реализация были разработаны и отлажены Л. В. Городней. Л. В. Суковатицина (Черноброд) выполнила отладку Лисп-арифметики с целью поддержки будущих исследований в области верификации программ. Во время очередного визита в Новосибирск Джон Маккарти собственноручно написал тест для проверки функциональной полноты новосибирской реализации Лиспа. Система выдержала эту проверку.

## Применение Лисп-системы

Лисп-система на БЭСМ-6 эксплуатировалась практически без изменений почти до 1986 года (до смены элементной базы). О ней имеется косвенное упоминание в книге “Мир Лиспа”, один из авторов которой бывал в Новосибирске и очень дотошно вникал во все, относящееся к Лиспу и его применению. Мюнхенский музей Лиспа в качестве экспоната располагает публикацией о новосибирском Лисп-проекте. Основное применение системы — исследования в области верификации программ, эксперименты по организации недоопределенных вычислений и конструированию семантических моделей.

## Литтл-компилятор. Замысел реализации

Реализацию эффективной системы программирования для языка Set1 Дж. Шварц предполагал выполнить с помощью специально разработанного машинно-независимого машинно-ориентированного языка системного программирования Little, похожего на Fortran, но обрабатывающего произвольные коды и приспособленного к крупно блочной организации программ и данных. Предполагалось, что именно этот язык — подходящая основа для машинно-зависимого переноса программ на наши машины.

## Идеи построения переносимого компилятора

Советская разработка системы программирования и компилятора для языка Little на БЭСМ-6 была выполнена Л. В. Городней в 1976 году. При реализации этой системы на макроассемблере было предложено

и экспериментально опробовано комплексное решение ряда технологических проблем обеспечения машинно-зависимого переноса программ без кросс-трансляции. Решение базировалось на идеях вертикального расчленения программ, функциональной декомпозиции на простые классы семантических систем, синтаксическое и семантическое управление конструированием компилятора, виртуального кода программы и оптимизирующей кодогенерации.

### **Вспомогательные инструменты**

Попутно, чтобы читать программы с американских перфокарт, которые в отличие от наших (до появления ЕС ЭВМ) воспринимались узкой стороной, а не широкой, как на БЭСМ-6, был сделан перекодировщик (драйвер, как теперь говорят), отладка которого по сути дела была осуществлена с привлечением американских коллег в режиме почты того времени, т.е. две недели туда, две недели обратно в лучшем случае.

Заодно был разработан специальный формат листинга для компактной распечатки машинного кода с идентификацией точек входа, что было немаловажно, т.к. расход бумаги планировался на месяц вперед и лимит был достаточно жестким.

Для этого компилятора Л. А. Степанова реализовала условный макроассемблер, обеспечивающий интегрированное хранение машинно-зависимого текста программы для набора разных архитектур. Н. Н. Зыков, Н. П. Шиян и Н. А. Шупта реализовали три разных текстовых макрогенератора.

### **Сухой остаток**

Опыт этой разработки дал материал для формализации реализации семантики языков программирования в виде интерпретируемых сетей, допускающих сетевое определение правила их функционирования. Кроме того, была выполнена систематизация требований ЕСПД к программной документации и представлена в форме, сходной с гипертекстом.