***Тема 1.3. Структура математических пакетов прикладных программ и их основные компоненты***

[**1.3.1. Структуры ППП**](#_1.2.1._Структуры_ППП)

[**1.3.2. Эволюция ППП. Современные ППП**](#_1.2.2._Эволюция_ППП.)

### 1.3.1. Структуры ППП

Итак, ***пакет прикладных программ (ППП)*** – это комплекс взаимосвязанных программ для решения определенного класса задач из конкретной предметной области. На текущем этапе развития информационных технологий именно ППП являются наиболее востребованным видом прикладного обеспечения. Это связано с упомянутыми ранее особенностями ППП. Рассмотрим их подробней:

* ***Ориентация на решение класса задач****.* Одной из главных особенностей является ориентация ППП не на отдельную задачу, а на некоторый класс задач, в том числе и специфичных, из определенной предметной области. Так, например, офисные пакеты ориентированы на офисную деятельность, одна из задач которой - подготовка документов (в общем случае включающих не только текстовую информацию, но и таблицы, диаграммы, изображения). Следовательно, офисный пакет должен реализовывать функции обработки текста, иметь средства обработки табличной информации, средства построения диаграмм разного вида и первичные средства редактирования растровой и векторной графики.
* ***Наличие языковых средств****.* Другой особенностью ППП является наличие в его составе специализированных языковых средств, позволяющих расширить число задач, решаемых пакетом или адаптировать пакет под конкретные нужды. Пакет может представлять поддержку нескольких входных языков, поддерживающих различные парадигмы. Поддерживаемые языки могут быть использованы для формализации исходной задачи, описания алгоритма решения и начальных данных, организации доступа к внешним источникам данных, разработки программных модулей, описания модели предметной области, управления процессом решения в диалоговом режиме и других целей. Примерами входных языков ППП являются VBA в пакете MS Office, AutoLISP/VisualLISP в AutodeskAutoCAD, StarBasic в OpenOffice.org
* ***Единообразие работы с компонентами пакета.*** Еще одна особенность ППП состоит в наличии специальных системных средств, обеспечивавших унифицированную работу с компонентами. К их числу относятся специализированные банки данных, средства информационного обеспечения, средства взаимодействия пакета с операционной системой, типовой пользовательский интерфейс и т.п.

Несмотря на разнообразие конкретных пакетных разработок, их обобщенную внутреннюю структуру можно представить в виде трех взаимосвязанных элементов (рис. 1.2-1):

1. входной язык (макроязык, язык управления) - представляет средство общения пользователя с пакетом;
2. предметное обеспечение (функциональное наполнение) - реализует особенности конкретной предметной области;
3. системное обеспечение (системное наполнение) - представляет низкоуровневые средства, например, доступ к функциям операционной системы.

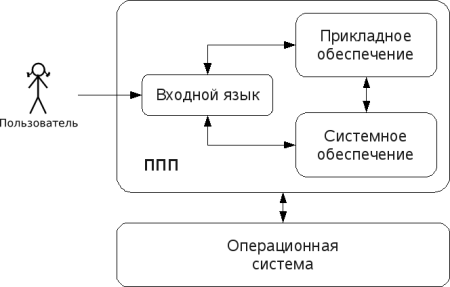


Рис. 1.2-1. Структура ППП

***Входной язык*** - основной инструмент при работе пользователя с пакетом прикладных программ. В качестве входного языка могут использоваться как универсальные (Pascal, Basic и т.п.), так и специализированные, проблемно-ориентированные языки программирования (Cobol - для бизнес-приложений, Lisp - списочные структуры данных, Fortran и MathLAB - математические задачи и т.п.).

Развитый пакет может обладать несколькими входными языками, предназначенными для выполнения различных функций в рамках решаемого класса задач. Так, например в пакете OpenOffice.org поддерживаются языки StarBasic, Python, JavaScript и Java. StarBasic является основным входным языком, предназначенным для автоматизации работы с пакетом, для этого языка имеется интегрированная среда разработки и встроенный отладчик. Скрипты на языках Python и JavaScript загружаются и исполняются из внешних файлов. На Java (через SDK и функции API OpenOffice) можно создавать модули расширения и полнофункциональные приложения-компоненты.

Входные языки отражают объем и качество предоставляемых пакетом возможностей, а также удобство их использования. Таким образом, именно входной язык является основным показателем возможностей ППП. Однако стоит отметить, что в современных пакетах обращение пользователя к языковым средствам обычно происходит косвенно, через графический интерфейс.

***Предметное обеспечение*** отражает особенности решаемого класса задач из конкретной предметной области и включает:

* программные модули, реализующие алгоритмы (или их отдельные фрагменты) прикладных задач;
* средства сборки программ из отдельных модулей.

Наиболее распространено в настоящее время оформление программных модулей в виде библиотек, подключаемых статически или динамически. В зависимости от использованного разработчиками подхода к проектированию и реализации ППП такие библиотеки содержат встроенные классы и описания их интерфейсов (при использовании объектно-ориентированного программирования). При использовании парадигмы структурного программирования в библиотечных модулях содержатся процедуры и функции, предназначенные для решения некоторых самостоятельных задач. В обоих случаях библиотеки связаны с другими модулями пакета лишь входной и выходной информацией.

***Системное обеспечение*** представляет собой совокупность низкоуровневых средств (программы, файлы, таблицы и т.д.), обеспечивающих определенную дисциплину работы пользователя при решении прикладных задач и формирующих окружение пакета. К системному обеспечению ППП относят следующие компоненты:

* монитор - программа, управляющая взаимодействием всех компонентов ППП;
* транслятор(ы) с входных языков - для ППП характерно использование интерпретируемых языков;
* средства доступа к данным - драйверы баз данных и/или компоненты, представляющие доступ через унифицированные интерфейсы (ODBC, JDBC, ADO, BDE и т.п.);
* информационно-справочный модуль - предоставляет функции поддержки, среди которых информационные сообщения, встроенная справочная системы и т.п.
* различные служебные программы, выполняющие низкоуровневые операции (автосохранение, синхронизация совместно используемых файлов и т.д.)

Приведенная логическая структура ППП достаточно условна и в конкретном ППП может отсутствовать четкое разделение программ на предметное и системное обеспечение. Например, программа планирования вычислений, относящаяся к прикладному обеспечению, может одновременно выполнять и ряд служебных функций (информационное обеспечение, связь с операционной системой и т.п.).

Кроме того, одни и те же программы в одном пакете могут относиться к предметному обеспечению, а в другом - к системному. Так, программы построения диаграмм в рамках специализированного пакета машинной графики естественно отнести к предметному обеспечению. Однако те же программы следует считать вспомогательными и относящимися к системному обеспечению, например, в пакете решения вычислительных задач.

### 1.3.2. Эволюция ППП. Современные ППП

Первые ППП представляли собой простые тематические подборки программ для решения отдельных задач в той или иной прикладной области, обращение к ним выполнялось с помощью средств оболочки ОС или из других программ. Современный пакет является сложной программной системой, включающей специализированные системные и языковые средства. В относительно короткой истории развития вычислительных ППП можно выделить *4 основных поколения* (класса) пакетов. Каждый из этих: классов характеризуется определенными особенностями входящих состав ППП компонентов - входных языков, предметного и системного обеспечения.

**Первое поколение**. В качестве входных языков ППП первого поколения использовались универсальные языки программирования (Фортран, Алгол-60 и т.п.) или языки управления заданиями соответствующих операционных систем. Проблемная ориентация входных языков достигалась за счет соответствующей мнемоники в идентификаторах. Составление заданий на таком языке практически не отличалось от написания программ на алгоритмическом языке. Предметное обеспечение первых ППП, как правило, было организовано в форме библиотек программ, т.е. в виде наборов (пакетов) независимых программ на некотором базовом языке программирования (отсюда впервые возник и сам термин «пакет»). Такие ППП иногда называют пакетами библиотечного типа, или пакетами простой структуры. В качестве системного обеспечения пакетов первого поколения обычно использовались штатные компоненты программного обеспечения ЭВМ: компиляторы с алгоритмических языков, редакторы текстов, средства организации библиотек программ, архивные системы и т.д. Эти пакеты не требовали сколько-нибудь развитой системной поддержки, и для их функционирования вполне хватало указанных системных средств общего назначения. В большинстве случаев разработчиками таких пакетов были прикладные программисты, которые пытались приспособить универсальные языки программирования к своим нуждам.

**Второе поколение**. Разработка ППП второго поколения осуществлялась уже с участием системных программистов. Это привело к появлению специализированных входных языков на базе универсальных языков программирования. Проблемная ориентация таких языков достигалась не только за счет использования определенной мнемоники, но также применением соответствующих языковых конструкций, которые упрощали формулировку задачи и делали ее более наглядной. Транслятор с такого языка представлял собой препроцессор (чаще всего макропроцессор) к транслятору соответствующего алгоритмического языка. В качестве модулей в пакетах этого класса стали использоваться не только программные единицы (т.е. законченные программы на том или ином языке программирования), но и такие объекты, как последовательность операторов языка программирования, совокупность данных, схема счета и др. Существенные изменения претерпели также принципы организации системного обеспечения ППП. В достаточно развитых пакетах второго поколения уже можно выделить элементы системного обеспечения, характерные для современных пакетов: монитор, трансляторы с входных языков, специализированные банки данных, средства описания модели предметной области и планирования вычислений и др.

**Третье поколение.** Третий этап развития ППП характеризуется появлением самостоятельных входных языков, ориентированных на пользователей-непрограммистов. Особое внимание в таких ППП уделяется системным компонентам обеспечивающим простоту и удобство. Это достигается главным образом за счет специализации входных языков и включения в состав пакета средств автоматизированного планирования вычислений.

**Четвертое поколение**. Четвертый этап характеризуется созданием ППП, которые эксплуатируются в интерактивном режиме работы. Основным преимуществом диалогового взаимодействия с ЭВМ является возможность активной обратной связи с пользователем в процессе постановки задачи, ее решения и анализа полученных результатов. Появление и интенсивное развитие различных форм диалогового общения обусловлено, прежде всего, прогрессом в области технических средств (графическая подсистема ЭВМ и средства мультимедиа, сетевые средства). Развитие аппаратного обеспечения повлекло за собой создание разнообразных программных средств поддержки диалогового режима работы (диалоговые операционные системы, диалоговые пакеты программ различного назначения и т. д.). Прикладная система состоит из диалогового монитора - набора универсальных программ, обеспечивающих ведение диалога и обмен данными, и базы знаний об области. Информация о структуре, целях и форма диалога задает сценарий, в соответствии с который монитор управляет ходом диалога. Носителями процедурных знаний о предметной области являются прикладные модули, реализующие функции собственной системы. Таким образом, создание прикладной системы сводится к настройке диалогового монитора на конкретный диалог, путем заполнения базы знаний. При этом программировать в традиционном смысле этого слова приходится лишь прикладные модули, знания о диалоге вводятся в систему с помощью набора соответствующих средств - редактора сценариев. Логично требовать, чтобы редактор сценариев также представлял собой диалоговую программу, отвечавшую рассмотренным выше требованиям. Благодаря готовому универсальному монитору программист может сосредоточиться на решении чисто прикладных задач, выделение же знаний о диалоге в сценарий обеспечивает в значительной степени необходимая гибкость программного продукта. Большое внимание в настоящее время уделяется проблеме создания «интеллектуальных ППП». Такой пакет позволяет конечному пользователю лишь сформулировать свою задачу в содержательных терминах, не указывая алгоритма ее решения. Синтез решения и сборка целевой программы производятся автоматически. При этом детали вычислений скрыты от пользователя, и компьютер становится интеллектуальным партнером человека, способным понимать его задачи. Предметное обеспечение подобного ППП представляет собой некоторую базу знаний, содержащую как процедурные, так и описательные знания. Такой способ решения иногда называют концептуальным программированием, характерными особенностями которого является программирование в терминах предметной области использование ЭВМ уже на этапе постановки задач, автоматический синтез программ решения задачи, накопление знаний о решаемых задачах в базе знаний.

Для иллюстрации ранее рассмотренных материалов приведем несколько примеров современных пакетов прикладных программ из различных предметных областей. Учитывая, что постоянно появляются новые версии программных продуктов, здесь будут рассматриваться не возможности конкретных версий, а лишь основные структурные компоненты, входящие в состав того или иного пакета.

**AutodeskAutoCAD**. Основное назначение ППП AutoCAD - создание чертежей и проектной документации. Современные версии этого пакета представляют значительно большие возможности, среди которых построение трехмерных твердотельных моделей, инженерно-технические расчеты и многое другое. Первые версии системы AutoCAD, разрабатываемой американской фирмой Autodesk, появились еще в начале 80-х годов двадцатого века, и сразу же привлекли к себе внимание своим оригинальным оформлением и удобством для пользователя. Постоянное развитие системы, учет замечаний, интеграция с новыми продуктами других ведущих фирм сделали AutoCAD мировым лидером на рынке программного обеспечения для автоматизированного проектирования.

***Языковые средства***. В основе языковых средств ППП AutoCAD - технология Visual LISP, базирующаяся на языке AutoLISP (подмножество языка LISP) и используемая для создания приложений и управления в AutoCAD. Visual LISP представляет полное окружение, включающее:

* Интегрированную среду разработки, облегчающую написание, отладку и сопровождение приложений на AutoLISP
* Доступ к объектам ActiveX и обработчикам событий
* Защиту исходного кода
* Доступ к файловым функциям операционной системы
* Расширенные функции языка LISP для обработки списочных структур данных.

Для разработчиков совместимых приложений в AutoCAD включена поддержка ObjectARX. Это программное окружение представляет объектно-ориентированный интерфейс для приложений на языках C++, C

***Основными тенденциями в развитии ППП*** на сегодняшний день в качестве основных факторов, влияющих на функциональность ППП и сложность их разработки ПО, можно отметить следующие:

* рост производительности персональных компьютеров;
* расширение классов решаемых задач;
* увеличение общего числа пользователей;
* значительное количество ранее созданного (наследованного) ПО;
* развитие Интернет и корпоративных сетей.

Разработка приложений с учетом этих факторов привела к появлению прикладных пакетов и интегрированных сред, которые по своим характеристикам выходят за рамки ППП четвертого поколения. Среди отличительных черт ПОнового поколения выделяют следующие:

* интеграция компонентов прикладного пакета не только с приложениями пакета, но и с окружением;
* широкое использование отраслевых стандартов;
* использование инфраструктуры Интернет;
* платформ независимость.

Особую значимость на дальнейший сценарий развития ППП имеет влияние технологий Интернет и, в частности, Web. Возможности, предоставляемые глобальной сетью, позволяют обмениваться любой информацией, которую можно представить в цифровом виде. Технологии Интернет уже сейчас с успехом используются в ведущих пакетах прикладных программ, в первую очередь для обеспечения совместной работы пользователей. Практическая реализация общего доступа возможна, например, с использованием промежуточного ПО (middleware). Так, при использовании технологии ActiveX в документ MS Word или таблицу MS Excel можно поместить любой документ, поддерживающий ActiveX. Внедренным может быть документ, размещенный в Интернет, более того, имеется потенциальная возможность отредактировать его и сохранить изменения в Сети. Все большей популярностью пользуется концепция “тонких клиентов”. Под “тонким клиентом” подразумевается Интернет-браузер. Современные браузеры позволяют отображать не только гипертекстовые документы, но и изображения в растровых и векторных форматах, видео- и аудиоданные. Кроме этого, браузеры предоставляют средства интерактивного взаимодействия с веб-серверами в виде различных веб-форм (от форм авторизации или поиска до форм загрузки файлов) и поддерживают выполнение программ-скриптов на своей стороне. Это позволяет создавать программы, загружаемые с веб-сервера, но выполняемые в браузере. Примером такого решения являются сервисы GoogleDocs (Google Документы). Пользователям этого сервиса предоставляется возможность создавать и редактировать текстовые документы, электронные таблицы и презентации прямо в окне браузера, сохранять их в Интернет и предоставлять в совместное использование.

***Облачные технологии.*** Ввиду повсеместного проникновения Интернета, можно говорить о том, что прикладное программное обеспечение будет переходить в разряд сервиса, то есть пользователи будут работать с необходимым программным обеспечением через Сеть, получая на свои компьютеры готовые результаты. Следовательно, необходимость в больших локальных мощностях частично отпадет, что будет способствовать росту спроса на недорогие компьютеры с низким энергопотреблением.

В основе технологий, которые обеспечивают подобные возможности, лежит ряд совместных наработок ведущих производителей ПО и организаций по стандартизации ПО. К ним относятся сервисно-ориентированная архитектура корпоративных приложений (веб-сервисы) и стандартизованные форматы документов.

***Веб-сервис*** или ***веб-служба***, (англ. webservice) - это программная система, доступная через локальную или глобальную сеть по заданному адресу, чьи общедоступные интерфейсы определены на языке XML. Эта программная система доступна другим программными системами, которые могут взаимодействовать с ней посредством XML-сообщений. передаваемых с помощью ***интернет-протоколов***. Веб-служба является единицей модульности при использовании сервисно-ориентированной архитектуры приложения. Иными словами, веб-сервис - это именованный компонент распределенной прикладной системы, доступный по гипертекстовым протоколам (HTTP, HTTPS и подобным).

Сервисно-ориентированные приложения построены на следующих индустриальных стандартах:

* **XML** – Расширяемый язык разметки, предназначенный для хранения и передачи структурированных данных;
* **SOAP** – Протокол обмена сообщениями на базе XML;
* **WSDL** – Язык описания внешних интерфейсов веб-службы на базе XML;
* **UDDI** – Универсальный интерфейс распознавания, описания и интеграции (UniversalDiscovery, Description, andIntegration). Каталог веб-служб и сведений о компаниях, предоставляющих веб-службы во всеобщее пользование или конкретным компаниям.

**Основными достоинствами** веб-сервисов являются:

* интероперабельность;
* открытость архитектуры;
* взаимодействие программных систем через средства защиты информации (прокси-серверы, межсетевые экраны).

**Основным недостатком** является меньшая производительность приложений и больший объем сетевого трафика по сравнению с другими технологиями распределенных вычислений (RMI, CORBA, DCOM/ActiveX). Еще одним недостатком является повышенная требовательность к аппаратным ресурсам на стороне сервера приложений (поставщика веб-сервисов).

Перспективным направлением в развитии ППП является использование унифицированных форматов документов на основе ***открытых стандартов***. Открытый стандарт - общедоступная спецификация, обычно разрабатываемая некоммерческой организацией по стандартизации, свободная от лицензионных ограничений при использовании. Открытые форматы являются подмножеством открытых стандартов и определяют спецификации хранения и представления цифровых данных. Использование открытых форматов в ППП позволяет гарантировать возможность доступа к данным из любого совместимого приложения без оглядки на лицензионные права и технические спецификации. Актуальность концепции открытых форматов подтверждается практикой - правительственные организации многих стран используют их в качестве основного средства.

На сегодняшний день разработаны и применяются открытые форматы практически для всех классов задач, решаемых ППП, начиная от офисных приложений до мультимедийных данных и 3D-графики.

OpenDocumentFormat (ODFот OASISOpenDocumentFormatforOfficeApplication– открытый формат документов для офисных приложений) - открытый формат файлов документов для хранения и обмена редактируемыми офисными документами: текстовыми документами, электронными таблицами, рисунками, базами данных, презентациями.

Стандарт был разработан индустриальным сообществом OASIS и основан на XML-формате. 1 мая 2006 года принят как международный стандарт ISO/IEC 26300, доступен для всех и может быть использован без ограничений. Этот формат поддерживается в таких ППП как OpenOffice.org, IBM LotusSymphony, Koffice, Scribus, GoogleDocs, AjaxWrite, MicrosoftOffice 2007 SP2.

Распространение веб-сервисов на основе открытых стандартов ведет к ситуации, когда вместо запускa определенных программ корпоративные пользователи смогут получить доступ к любым прикладным средствам, необходимым в данный момент, просто подключившись к Интернет. В таком контексте приложения могут быть представлены как свободно и бесплатно, так и платно, по подписке, в зависимости от объема потребления. Сочетание широкополосного интернета с платформонезависимыми приложениями (написанными, к примеру, на языке Java) в некоторых областях уже сделали модель коммунальных услуг в области ИТ реальностью. Например, Salesforce.com за умеренную месячную плату предлагает в интернете приложения для управления отношениями с клиентами (CRM). Пользователям, количество которых уже составляет около 100 тыс., не нужно устанавливать или поддерживать у себя сложные пакеты CRM. Им достаточно только запустить браузер и подключиться к сервeрaм и услугам Salesforce.com. В свою очередь свободный доступ к офисным приложениям представляет ранее упомянутый сервис GoogleDocs, число пользователей которого составляет более полумиллиона.

Одним из наиболее широких направлений внедрения информационных технологий в среднем профессиональном образовании является применение прикладного программного обеспечения.

ППП - это комплекс взаимосвязанных программ для решения задач определенного класса. В соответствии с применяемой классификацией прикладного программного обеспечения его можно разделить на такие основные группы:

* проблемно-ориентированные программные продукты и пакеты;
* системы автоматизированного проектирования (САПР);
* методы-ориентированные пакеты прикладных программ;
* пакеты прикладных программ общего назначения;
* интеллектуальные информационные системы;
* офисные пакеты прикладных программ;
* программные средства мультимедиа;
* настольные издательские системы.

***Проблемно-ориентированные программные продукты и пакеты*** – это обширная группа пакетов программ, разработанных для автоматизации процессов решения различных функциональных задач, какв промышленной, так и в непромышленной сферах деятельности человека.

***Системы автоматизированного проектирования*** - представляет собой организационно-техническую систему, предназначенную для автоматизации процесса проектирования, состоящую из персонала и комплекса технических, программных и других средств автоматизации его деятельности.

***Методо-ориентированные пакеты прикладных программ -*** включает программные продукты, обеспечивающие независимо от предметной области и функции информационных систем математические, статистические и другие методы решения задач.

***Пакеты прикладных программ общего назначения -*** Поддерживают компьютерные технологии конечных пользователей и включают текстовые и табличные процессоры, графические редакторы, системы управления базами данных (СУБД).

***Интеллектуальные информационные системы*** - представляет собой комплекс программных, лингвистических и логико-математических средств для реализации основной задачи: осуществление поддержки деятельности человека, например возможность поиска информации в режиме продвинутого диалога на естественном языке.

***Офисные пакеты прикладных программ -*** обеспечивают организационное управление деятельностью офиса. Включают органайзеры (записные и телефонные книжки, календари, презентации и т.д.), средства распознавания текста.

***Программные средства мультимедиа -*** комплекс аппаратных и программных средств, позволяющих пользователю работать в диалоговом режиме с разнородными данными (графика, текст, звук, видео), организованными в виде единой информационной среды.