

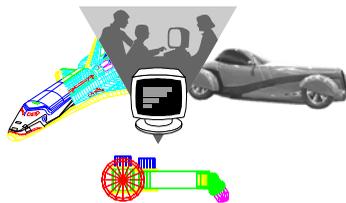
საქართველოს ტექნიკური უნივერსიტეტი

მაგისტრატურის დეპარტამენტი,
მანქანათმშენებლობის ტექნოლოგიური პროცესების
ოპტიმიზაციის ლაბორატორია,
CAD/CAM-ის ჯგუფი

ParametricCAD'97

მოსხენებათა კრებული

(სამეცნიერო კონფერენციის მოხსენებათა
კრებული, 3-4 ივლისი, 1008 წელი)



თბილისი 1008 წელი

Исследование объектно-ориентированной архитектуры САПР-ТП

аспирант Л. Мегрелишвили

Традиционными задачами проектирования являются задачи синтеза и анализа. При этом под синтезом понимается процесс принятия решения, а под анализом процесс оценки принятого решения.

САПР(Система автоматизированного проектирования) - это человеко-машинная система, которая решает задачи автоматизированного синтеза и автоматизированного анализа. В процессе взаимодействия со средствами САПР в интерактивном режиме проектировщик осуществляет получение проектного решения на основе моделей синтеза и анализа. Модель - это формализованное описание знаний, при помощи которых система осуществляет интерпретацию и формирование решений.

Термин САПР является смысловым аналогом - эквивалентом английского CAD (сокращено Computer Aided Designing) и означает проектирование с помощью компьютера, человеку в этом процессе отводится активная роль.

САПР - состоит из комплекса средств автоматизированного проектирования (КСАП), взаимосвязанного с необходимыми подразделениями проектной организации.

САПР - это система проектирования, в которой органично объединены творческие усилия коллектива проектировщиков и возможности математических методов и компьютерных систем на всей совокупности взаимосвязанных этапов проектирования с применением развитых средств программного и информационного обеспечения для коренного улучшения качества проектных работ и сокращения их сроков, что достигается благодаря :

- *систематизации и совершенствованию проектного процесса, которые сопровождаются перестройкой структуры и кадрового состава проектных организаций;*
- *применению эффективных математических*

моделей проектируемых объектов;

- комплексной оптимизации принимаемых решений;
- улучшению информационного обеспечения разработок;
- автоматизации трудоемких рутинных работ;
- частичной замене макетирования.

Можно выделить два основных направления исследований в разработке САПР:

1. Изучение методологии автоматизированного проектирования ;

2. Развитие новых методов организации и разработки самих САПР .

Первое направление связано с изучением теоретического, аксиоматического и алгоритмического аспектов САПР, т.е. с разработкой технологии проектирования как науки, второе - с разработкой теоретических и практических методов реализации самих систем в рамках уже известных технологических подходов, достаточно хорошо зарекомендовавших себя в условиях промышленного применения.

Одними из основных принципов , на которых основаны промышленные САПР это:

Концептуальность.

В основу разработки положена логически завершенная система взглядов, определяющих оригинальные научно - методические положения по комплексной автоматизации конструкторско - технологического проектирования в условиях мелкосерийного и серийного машиностроения.

Комплексность.

Функциональная структура системы ориентирована на подготовку комплекта технологической документации по вновь осваиваемому изделию или для его технологического сопровождения в производстве.

Универсальность.

Концептуальный, функциональный, программный, технический и организационный комплексы системы не имеют каких - либо принципиальных зависимостей от специфики эксплуатирующих ее предприятий. Собственно проектирование технологических процессов осуществляется с использованием различных форм

взаимодействия человека с компьютерной системой. Относительная простота - она отражается в методиках создания и развития программно - информационного комплекса системы, а также быстрой адаптации к системе ее непосредственного пользователя.

Интегрируемость.

Используемой концепцией предусмотрена возможность сквозного и произвольного комплексирования человеко - машинных функций системы как по горизонтали (равнопочиненные функции), так и по вертикали (соподчиненные функции) без необходимости в каких либо специальных программно - информационных блоках сопряжения.

Открытость.

Если раньше автоматизированные процессы проектирования создавались для решения проблем, ориентированных на изделие или учитывающих специфику производства то последние разработки свидетельствуют о все увеличивающейся переориентации на создание универсальных методик и решения задач выходящих за пределы отдельных отраслей производства. Цель такой переориентации - пригодность отдельных систем для различных отраслей. Универсализация базовых систем обуславливает снижение расходов пользователя. Поэтому это свойство САПР обусловлено необходимостью достижения гибкости системы, которое является одним из важных свойств САПР.

Гибкость САПР с точки зрения расширения возможностей ее использования может быть увеличена, если программное обеспечение САПР является универсальным и открытым, в основу структуры таких систем положен принцип - открытой архитектуры, позволяющий адаптировать и развивать многие функции применительно к конкретным задачам и требованиям конечного пользователя.

Каждая САПР имеет свое целевое назначение, т.е. предназначена для решения определенного класса прикладных задач конечного пользователя. Поэтому одной из важных характеристик функционирования САПР является способность системы адаптироваться на

конкретные задачи конечного пользователя. Аналогичный вывод сделан американским аналитиком в области САД/САМ Ральфом Гробовским [3], по сведениям которого начиная с 1993 года ведущие производители систем САД/САМ основным требованием к собственной продуктам ставят обеспечение свойства 'используемости' ('Usability'), что означает способность системы адаптироваться на конкретные задачи конечного пользователя. По свойству адаптируемости системы можно разделить на два класса: системы с жесткой архитектурой и системы с открытой архитектурой.

В системах с закрытой архитектурой пользователь не имеет возможности модифицировать программно-математическое обеспечение и адаптация происходит на уровне файлов базы данных посредством заполнения соответствующей информации, т.е. конечный пользователь может адаптировать систему только на уровне информационного обеспечения.

В системах с открытой архитектурой у пользователя есть возможность использования собственных решений не только на уровне информационного обеспечения, но и на уровне функционирования самой системы посредством изменения программно-математического обеспечения. Такие системы дают более широкие возможности по адаптации на конкретные задачи конечного пользователя, т.к не требуют вмешательства со стороны разработчика системы.

Степень вмешательства пользователя в виды обеспечения КСАП пропорциональна способности системы адаптироваться на задачи конечного пользователя.

По степени открытости можно выделить следующие поколения систем с открытой архитектурой :

1-е поколение - системы, имеющие настраиваемую конфигурацию. Такие системы строятся по модульному принципу. При этом выделяются отдельно программируемые модули, где пользователь имеет возможность сформировать конфигурацию системы путем выбора отдельных модулей, т.е осуществить адаптацию программного и технического обеспечения. Основным

элементом в таких системах является модуль, который должен удовлетворять требованиям универсальности и информативности. Информационное согласование с другими модулями осуществляется с помощью специальных программ- драйверов, а для информационной согласованности технических модулей используются специальные устройства-адаптеры. Типичными представителями такой системы является САПР электронных схем PCAD .

2-е поколение - системы, в которых адаптация на конкретные задачи конечного пользователя происходит благодаря функционированию системы на основе пользовательских моделей принятия решения. Типичными представителями таких систем являются экспертные системы. Структурно экспертные системы состоят из двух частей:

1. Машина логических выводов.
2. База знаний.

Искусственный интеллект (ИИ) - это программная система, имитирующая на компьютере мышление человека. Для создания такой системы необходимо изучить процесс мышления человека, решающего определенные задачи или принимающего решения в конкретной области, выделить основные шаги этого процесса и разработать программные средства, воспроизводящие их на компьютере. Система ИИ, созданная для решения задач в конкретной проблемной области, называется экспертной системой (ЭС). Источником знаний для наполнения экспертных систем служат эксперты в соответствующей предметной области. Работа всех экспертных систем основана на экспертной информации, полученной в конкретной предметной области.

Одним из путей улучшения характеристик по производительности и надежности, сократить сроки исследований и разработок новых видов продукции машиностроения, является создание проблемно - ориентированных экспертных систем.

Важным фактором, определяющим возможность использования ЭС, особенно в сфере проектирования, следует считать развитие взаимосвязи между ЭС, и



И-1

традиционным программным обеспечением. Способность ЭС, взаимодействовать с системами разработки, анализа, базами данных является необходимой для ее успешного функционирования. Целесообразно установление обратной связи ЭС с пользователем для дополнения базы знаний. Возможность расширения базы знаний - главное преимущество ЭС.

При такой архитектуре разработчиками САПР разрабатывается машина логических выводов, а база знаний формируется конечным пользователем, тем самым осуществляется проблемная ориентация системы. Т.о. экспертные системы позволяют адаптироваться на уровне моделей принятия решений.

Развитие ЭС зависит от решения таких проблем, как представление знаний, разработка функциональных языков и машинных способов извлечения знаний, обобщение на естественном языке, построение прототипов ЭС.

Совершенствование ЭС в значительной мере связано с успехами в разработке человеко-машинного интерфейса, технического и программного обеспечения ЭС.

3-е поколение - компилируемые системы- системы с возможностью компиляции исполняемых модулей, что

позволяет пользователю адаптировать системы на уровне программно-математического обеспечения, т.е. в таких системах у пользователя есть возможность изменить исходный вариант программно-математического обеспечения.

Как обычно, такие системы делятся на ядро прикладных модулей, которыми решаются основные прикладные задачи и интерфейсные модули, осуществляющие связь прикладных модулей с периферийными устройствами ($hbc=1$). Следовательно, по степени "вмешательства" пользователя в модули системы можно выделить следующие классы систем:

1. Системы с возможностью модификации только на уровне интерфейсных модулей без ядра прикладных модулей. К таким системам можно отнести системы визуального программирования *Delphi, Visual C, Visual Basic, среда Windows*.

2. Системы, в которых присутствовало ядро прикладных задач и вмешательство пользователя осуществлялось только на уровне обеспечения каналов интерфейса. Типичными представителями является *Mini Studio*, в котором пользователь имеет возможность создавать собственные драйверы для принтера и плотера, осуществить настройку интерактивного режима с пользователем.

3. Системы, позволяющие вмешиваться не только на уровне каналов интерфейса, но и на уровне ядра решения прикладных задач. Типичным представителем является пакет *CADKey*.

В статье Ральфа Гробовского "Наступающая революция в CAD", подчеркнута важность двух терминов "Usability" - используемость и "Customizable" - адаптируемость.

Основной смысл, заключенный в этих терминах является следующее: вместо того чтобы покупать полный пакет *CAD Key*, конечный пользователь может гораздо дешевле выбрать и купить нужные ему приложения в виде отдельных *DLL* - файлов. Кроме этого посредством добавленных новых функций, конечный пользователь может осуществить настройку интерфейса, полностью или частично изменить внешний вид функционирования

системы, собранной из выбранных пользователем DLL - файлов, имеются также доступы по всем вызовам системы Windows, в которой функционирует CAD - пакет. Помимо этого возможно осуществить настройку драйверов плотера, настроить функционирование CAD - пакета с базой данных. CAD - пакет имеет 3D - систему.

CAD Key - это пакет с использованием COD (CAD Key Object Developer) технологии, обеспечивает ядро графического процессора. Это объектно - ориентированный продукт, созданный на языке C++ и поддерживаемый DXF и DWG форматами и системами Windows и Windows NT.

Корпорация Bentley Systems (Экстон, штат Пенсильвания) анонсировала очередную версию своей системы твердотельного моделирования Micro Station Modeler. В нее включен целый ряд новых функций, а кроме того, в этом продукте впервые реализована возможность "непрерывной модернизации". Пользователи, подписавшиеся на эту услугу, смогут сообщать о необходимых им усовершенствованиях через Web-страницу Bentley Select. Результаты разработок, выполненных по этим запросам, будут распространяться по Internet, а также на ежеквартально издаваемых компакт-дисках. В качестве отдельного продукта Bentley анонсировала Migration Package, с помощью которого обладатели пакета CADKEY смогут осуществить быстрый и дешевый переход на систему машиностроительного черчения Micro Station. Возможен также быстрый и дешевый переход с CADKEY на ПО твердотельного моделирования Micro Station Modeler. В состав обоих продуктов входит утилита DRAFT-PAK, предназначенная для преобразования файлов CADKEY в формат Micro Station.

Система Micro Station Modeler способна функционировать на многих платформах, включая ПК, под управлением Windows 3.x, Windows NT, а также машины Apple Power Macintosh, SunSPARC, HPRISC, SGI и DEC Alpha.

Важное место среди систем автоматизированного проектирования занимают системы автоматизированного проектирования технологических процессов (САПР

ТП). Объектом проектирования САПР ТП является сам технологический процесс, который можно представить в виде системы, которая по степени детализации описания свойств представляет собой иерархически - соподчиненную многоуровневую систему. Следовательно, проектирование такого объекта осуществляется в нескольких этапах и различными подсистемами. Например: САПР маршрутов, САПР отдельных операций, САПР заготовки, САПР инструментов и т.д.

Технологический процесс в случае ЧПУ (числовое программное управление) детализируется до уровня УП (управляющих программ) для станков с ЧПУ (устройством числового программного управления). Принципиальная особенность станка с ЧПУ заключается в том, что движение инструмента относительно обрабатываемой заготовки задается последовательностью положений и скоростями перемещений его рабочих органов, записываемых в числовой форме на специальном программном носителе.

Внедрение ЧПУ в технологию машиностроения обусловило необходимость построения числовых моделей технологического процесса, а следовательно и широкое использование математических методов и числовой вычислительной техники. Это привело к революционным изменениям в технологии машиностроения. Технология машиностроения из науки, носившей по преимуществу качественный характер, стала превращаться в науку точную. Технологический процесс обработки на станке с ЧПУ в отличие от традиционного технологического процесса требует большей детализации при решении технологических задач и учета специфики представления информации.

Задачи решаемые в САПР УП характеризуются двумя основными свойствами: оригинальность и многовариантность. Следовательно решение задачи в САПР УП должно осуществляться с учетом заранее подготовленных экспертных знаний, которые выражали бы опыт специалистов технологов.

Один из вопросов изучаемых в рамках развития новых методов организации и разработки САПР,

представляющих значительный научный и практический интерес - исследование способов представления в САПР ТП технологического опыта независимо от методов, лежащих в основе применяемого принципа проектирования. Любая система действует в соответствии с заложенными в нее алгоритмами проектирования, которые представляют собой элементы знаний специалистов в предметной области и являются формальным выражением их производственного опыта. Поскольку технологическое проектирование зачастую представляет собой набор плохо формализованных эвристик, которые в наибольшей степени подвержены влиянию традиций и нечетко определенной информации, то именно технологические знания, хранящиеся в системе претерпевают склонные изменения в течение ее жизненного цикла, а также при переносе ее с одного производства на другое. Было бы логично, учитывая специфику технологических знаний и необходимость их постоянное обновления использовать специфику и (как уже отмечалось выше) главное преимущество ЭС - возможность расширять базу знаний.

Однако недостаточный на сегодняшний день уровень формализации технологических задач, незавершенность технологических задач и теоретических основ процессов обработки, неполнота экспериментальных данных не позволяют полностью автоматизировать процессы подготовки технологий и УП для станков с ЧПУ.

Поэтому в большинстве случаев при подготовке УП оптимальной является работа в так называемом диалоговом режиме. В этом режиме наиболее формализованная часть технологических задач решается программно с использованием средств вычислительной техники, а дальнейшие пути решений на узловых участках проектирования выбирает технолог - программист. Это позволяет преодолеть недостаточную формализацию технологических задач, значительно сокращает время проектирования, однако требует творческого участия опытного знающего технолога.

Соответственно, особенностью использования ЭС в машиностроении является необходимость решения задач при наличии неформализованных данных и знаний, а также

в условиях лингвистической неопределенности. Кроме того, ЭС должны быть ориентированны на решение таких характерных для машиностроения задач, как диагностика, интерпретация данных, контроль, планирование, проектирование и управление.

Т.о. формализация знаний в ЭС подготовки УП является сложной и трудоемкой задачей, требующей от технолога высокой профессиональной подготовки не только в ряде технических дисциплин, но и в области математики и программной инженерии. Программной инженерией называют инженерно - научную дисциплину, предметом которой является то, что должны знать и уметь лица, занимающиеся различными видами инженерной деятельности по созданию и использованию ПО для профессионально качественного и высокопроизводительного выполнения своей работы.

Формализация вышеуказанных технологических алгоритмов для ЭС подготовки УП требует разработки программных модулей при помощи специальных языков программирования.

Языки описания технологических алгоритмов обладают достаточно ограниченными возможностями т.к. в них наблюдается отсутствие в большей или меньшей степени алгоритмической полноты. Это означает, что существует ряд классов проектных алгоритмов, нереализуемых в рамках таких языков, т.е. алгоритмов, которые невозможно смоделировать никакими комбинациями средств, имеющихся в данном языке.

Итак, с одной стороны мы имеем технологический процесс, который описывается сложным программно - математическим аппаратом, а с другой стороны язык программирования технологических алгоритмов, имеющий ограниченные алгоритмические возможности.

Поэтому при программировании УП возникают большие трудности, которые в конечном счете оказывают отрицательное влияние на качество производимой продукции, понижается качество и надежность программирования, увеличиваются сроки отладки программных модулей, что увеличивает время на технологическую подготовку производства.

Поэтому архитектура САПР NG должна иметь архитектуру экспертной системы, в базе знаний которой экспертные знания формализуются в виде технологических алгоритмов.

Наиболее перспективными в этом отношении представляются идеи абстрактных типов данных (АТД). Основные варианты были предложены во второй половине 70-х годов : (Хар, Лесков, Лондон, Парнас, Шо и др.), на которых основываются идеи объектно-ориентированного программирования (ООП).

Реализовать вышеописанную идею возможно при помощи идеологии ООП созданием системы объектов математических вычислений и отдельно -технологических модулей, при создании которых использовались бы указанные объекты в процессе своего функционирования .

Основные принципы объектно - ориентированной идеологии, это:

1. Инкапсуляция.
2. Наследование.
3. Подклассы.
4. Полиморфизм.

Инкапсуляция - объединение в одном элементе данных и процедур их обработки.

Наследование - сохранение, передача атрибутов данных и выполнение над ними операции от объекта к объекту.

Подклассы - все объекты создаются на основе классов и наследуют свойства и методы классов. Классы могут в свою очередь создаваться на основе других классов. Такие классы называются подклассами, они наследуют все свойства и методы класса. Подклассы позволяют нам сократить объем программирования и максимально использовать предыдущий опыт работы. Кроме того изменение свойств и методов родительского класса отслеживается и в подклассах. Т.о. можно без особых затрат изменить характеристики всего приложения.

Полиморфизм - в традиционном процедурном программировании аналогом является имя вызывающей подпрограммы.

Исходя из характера задач, решаемых в технологических модулях, и основываясь на объектно - ориентированной идеологии, мы должны определить классы объектов, методы и свойства объектов, характеризующих, описывающих и составляющих технологический процесс. Т.е. выделить составляющие технологического процесса,