

Ю. М. СОЛОМЕНЦЕВ В. Г. МИТРОФАНОВ В. В. ПАВЛОВ А. В. РЫБАКОВ

ИНФОРМАЦИОННО- ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ В МАШИНОСТРОЕНИИ CAE-ТЕХНОЛОГИИ



«НАУКА»

РОССИЙСКАЯ АКАДЕМИЯ НАУК
ИНСТИТУТ КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКОЙ
ИНФОРМАТИКИ

Ю. М. СОЛОМЕНЦЕВ В. Г. МИТРОФАНОВ
В. В. ПАВЛОВ А. В. РЫБАКОВ

И **Н** **Ф** **О** **Р** **М** **А** **Ц** **И** **О** **Н** **Н** **О** **В** **Ы** **Ч** **И** **С** **Л** **И** **Т** **Е** **Л** **Ь** **Н** **Ы** **Е** **С** **И** **С** **Т** **Е** **М** **Ы** **В** **М** **А** **Ш** **И** **Н** **О** **С** **Т** **Р** **О** **Е** **Н** **И** **С** **А** **С** **С** **-** **Т** **Е** **Х** **Н** **О** **Л** **О** **Г** **И** **И**



МОСКВА «НАУКА» 2003

УДК 681.518
ББК 32.965
И74

Ответственный редактор член-корреспондент Ю.М.Соломенцев

Рецензенты:

академик А.С.Бугаев;

д.т.н., профессор М.М.Аршанский, зав. кафедрой «Мехатроника
производственных систем» МГАПИ

Информационно-вычислительные системы в машиностроении CALS-технологии
/ Ю.М.Соломенцев, В.Г.Митрофанов, В.В.Павлов, А.В.Рыбаков – М.: Наука,
2003, 292 с.

ISBN 5-02-006261-8 (в пер.)

Эта книга о новых возможностях повышения эффективности промышленного производства, о совокупности методов и средств, моделях данных промышленной продукции, получивших название CALS-технологий. Все это создает новую среду деловых отношений и устанавливает стандарты и правила, которые позволяют преодолевать барьеры внутри и между предприятиями и организациями, найти общий язык всем деловым партнерам.

Для научных работников, инженеров, преподавателей и студентов технических вузов.

ISBN 5-02-006261-8

© Ю.М.Соломенцев, В.Г.Митрофанов,
В.В.Павлов, А.В.Рыбаков, 2003

© Издательство «Наука»
(художественное оформление), 2003

ОГЛАВЛЕНИЕ

| | |
|---|-----------|
| Введение | 5 |
| Глава 1. Концептуальный аспект CALS-технологий | 15 |
| 1.1. Основные концепции CALS-технологий | 15 |
| 1.2. Научные проблемы CALS-технологий | 19 |
| 1.3. Интегрированные производственные системы | 19 |
| 1.4. Интегрированное управление предприятием | 35 |
| 1.4.1. Базисная технология системы R/3 фирмы SAP | 35 |
| 1.4.2. Пример разработки модели интегрированного управления производством | 51 |
| 1.5. Взаимосвязь технологий и стандартов в области CALS | 60 |
| 1.6. Обучение CALS-технологиям | 68 |
| Литература | 73 |
| Глава 2. Математические модели в CALS-технологиях | 75 |
| 2.1. Структурное моделирование производственных систем | 75 |
| 2.1.1. Структурное моделирование в CALS-технологиях | 75 |
| 2.1.2. Структура полихроматического множества | 77 |
| 2.1.3. Основные операции над полихроматическими множествами | 99 |
| 2.1.4. Особенности полихроматических графов | 125 |
| 2.1.5. Структурное моделирование производственных систем в CALS-технологиях | 129 |
| 2.2. Моделирование технологических процессов | 135 |
| 2.2.1. Общие положения | 135 |
| 2.2.2. Основные математические понятия | 139 |
| 2.2.3. Таблицы принятия решений (таблицы соответствий) | 145 |
| 2.2.4. Выбор варианта базирования, последовательности обработки поверхностей и построение размерной структуры ТП | 152 |
| 2.2.5. Разложение графа на максимальные сильно связанные подграфы (приложение к задаче сборки или разузлования) | 162 |
| 2.2.6. Конечные автоматы (приложение к работе сборочного конвейера) | 167 |
| 2.2.7. Непрерывные модели | 171 |
| 2.2.8. Непрерывные оптимизационные задачи | 182 |
| Литература | 187 |

| | |
|--|------------|
| Глава 3. Опыт применения CALS-технологий | 188 |
| 3.1. Место и роль информатики в индустриальном обществе (на примере задач конструкторско-технологической подготовки производства) . . . | 197 |
| 3.1.1. Новое понимание реальности и существования в условиях информационных технологий | 201 |
| 3.2. Возможности компьютерно-технологической среды в ходе промышленного проектирования изделий машиностроения | 207 |
| 3.3. Направления развития автоматизированных систем при выпуске наукоемкой продукции в единичном и мелкосерийном производстве . . . | 213 |
| 3.3.1. Составление технического задания на систему | 216 |
| 3.3.2. Разбиение системы на компоненты | 219 |
| 3.3.3. Средства для макетирования технических систем | 219 |
| 3.3.4. Использование проектного опыта | 221 |
| 3.3.5. Повышение качества программ поиска информации | 223 |
| 3.3.6. Обратный инжиниринг | 225 |
| 3.4. Особенности анализа и проектирования деятельности в машиностроении при переходе к автоматизированным системам . . . | 227 |
| 3.4.1. Два подхода к технологии создания программного продукта | 227 |
| 3.4.2. Источники формирования требований к ПО | 230 |
| 3.4.3. Причины появления искажения требований к АС | 232 |
| 3.4.4. Аналитик по техническим процессам | 235 |
| 3.4.5. Архитектор АС | 237 |
| 3.4.6. Типовые подходы и типичные проблемы | 238 |
| 3.4.7. Сравнение основных моделей – плюсы и минусы | 241 |
| 3.4.8. Применение моделей | 243 |
| 3.4.9. Пояснения применяемой терминологии | 245 |
| 3.5. Возможности по созданию систем автоматизированной поддержки информационных решений силами пользователей | 246 |
| 3.5.1. Роль и место систем КТПП в обновлении выпускаемой продукции | 246 |
| 3.5.2. Ограничения традиционных систем КТПП | 247 |
| 3.5.3. Отличия «старого» и «нового» понимания КТПП | 249 |
| 3.5.4. Системы автоматизированной поддержки информационных решений | 251 |
| 3.5.5. Особенности построения САПИР | 252 |
| 3.6. Конструкторско-технологическая подготовка производства в машиностроении | 263 |
| 3.6.1. Сам себе программист | 264 |
| 3.6.2. КТПП как часть корпоративной информационной системы предприятия | 266 |
| 3.6.3. Проектирование на основе моделей | 266 |
| 3.6.4. Объекты проектирования на основе моделей | 269 |
| 3.6.5. Совмещенное проектирование | 270 |
| 3.6.6. Блочное-модульное проектирование | 272 |
| 3.6.7. «Разумные детали» и их коллективное использование | 272 |
| 3.6.8. Простота применения и изучения | 272 |
| 3.6.9. Решение задач КТПП в новых условиях | 273 |
| 3.7. Опыт использования информационных технологий при проектировании и изготовлении электродвигателей по индивидуальному заказу (на примере ОАО «Сафоновский электромашиностроительный завод») | 274 |
| Литература | 289 |

ВВЕДЕНИЕ

Развитие современной мировой экономики характеризуется усилением конкуренции на мировых рынках и ставит перед промышленниками и предпринимателями в качестве основной задачи экономии ресурсов (материальных, интеллектуальных, информационных и временных), привлекаемых для реализации конкретного проекта или программы на всех стадиях жизненного цикла изделий от разработки и производства до модернизации и утилизации.

Особую роль в решении этой группы проблем сыграли и продолжают играть информационные технологии, развитие которых можно охарактеризовать следующей динамикой:

- 1960 годы – автоматизация выполнения простейших функций;
- 1970 годы – интеллектуальная направленность информационных технологий, развитие информационного моделирования, прогнозирования и управления;
- 1980 годы – расширение областей применения информационных технологий, создание локальных сетей и электронных баз данных. Привлечение к использованию информационных технологий руководителей всех уровней управления;
- 1990 годы – стремление к объединению информационных ресурсов и кооперации при создании информационных технологий; совместное использование информации; создание виртуальных предприятий.

Таким образом, автоматизация технологических процессов и производств, основанная на информационных технологиях, является одним из главных средств подъема промышленного производства, создания конкурентоспособных изделий, продвижения продукции на внешний и внутренний рынки.

Глобальные процессы информатизации и компьютеризации открыли широкие возможности для повышения эффективности промышленного производства на базе параллельного проектирования, широкомасштабной кооперации и рациональной специализации, унификации и стандартизации проектной, производственной и эксплуатационной документации. Эти возможности нашли свое выражение в развивающейся совокупности методов и средств, получивших название CALS-технологий и представленных в серии международных стандартов STEP, P_LIB, MANDATE, EDIFACT и др.

Внедрение CALS-технологий в ближайшие годы станет необходимым условием выживания промышленных предприятий при существующей жесткой конкуренции товаров на международных и национальных рынках. Несоблюдение CALS-стандартов приведет при прочих равных условиях к заметному ухудшению потребительских свойств продукции, к увеличению себестоимости и сроков проектирования.

Книга вводит вас в CALS – новый способ выполнять работу. Название не должно вас беспокоить – оно будет объяснено ниже. Важно то, что под этим названием скрывается нечто, меняющее привычный способ работать. Это относится и к частному сектору, и к оборонному, и к государственному.

Эта книга о новых возможностях в работе, которые открывают перед нами новые информационные технологии.

Мы расскажем о том, что вы должны знать о новой технологии – о моделях данных о промышленной продукции, распределенных данных, об электронной коммерции.

Вам встретятся много сложного – это касается стандартов, обеспечения информационной безопасности и человеческого фактора. Но мы постараемся рассказать об этом простыми словами и ваши усилия не пропадут даром.

Эта книга не даст вам ответов на все вопросы, но она подскажет вам, где искать такие ответы.

Что такое CALS? CALS – это аббревиатура, которая сменила несколько значений. Первоначально это буквосочетание расшифровывалось как «компьютеризированные закупки и логистическая поддержка», затем «непрерывная поддержка закупок и жизненного цикла». Некоторые сегодня применяют расшифровку «коммерция со скоростью звука». Конкретное название не имеет большого значения, поскольку не меняется смысл того, что за ним кроется. А этот смысл касается людей, проектов, промышленной продукции, услуг и управления по новым правилам. Сегодняшние и будущие технологии, объединенные этим словом, позволяют работать быстрее, лучше и дешевле.

Использование этой технологии позволяет получить хорошую отдачу, обрести более широкие международные связи, повышает конкурентоспособность и эффективность производства товаров и услуг.

О каких технических приемах и какой технологии идет речь? Вы, возможно, слышали о них – об Интернете, Синтетической среде, Параллельном проектировании, Среде распределенных данных, Ре-инжиниринге бизнес процессов, CAD/CAM, Виртуальной реальности, Электронной коммерции. Они создают новую среду деловых отношений. В CALS создает схему, по которой все они могут работать вместе. Это означает установление новых стандартов и правил – в новой среде мы все должны говорить на одном языке.

Мы говорим о новой, улучшенной практике построения деловых отношений. Преодоление барьеров внутри и между различными организациями позволяет создавать виртуальные предприятия, участники которого могут работать вместе и быть при этом разделенными любыми расстояниями.

Оказывая людям поддержку в деловых отношениях, новая технология позволяет:

- создать информацию один раз и использовать ее всеми членами виртуального предприятия много раз, на всех стадиях жизненного цикла, от выработки концепции до утилизации;
- получить выгоду всем, вовлеченным в дело.

Технические термины и технологии

- ИНТЕРНЕТ/WWW;
- Синтетическая среда (SE);
- Системное проектирование;
- Управление требованиями;
- Управление данными о продукции (PDM);
- Параллельное проектирование (CE);
- Среда распределенных данных (SDE);
- Ре-инжиниринг бизнес процессов (BPRc);
- CAD/CAM автоматизированное проектирование/производство;
- Виртуальная реальность (VR);
- Электронная коммерция (e-commerce).

Немного истории. Идея родилась в оборонном секторе США и была названа CALS-стратегия Министерства обороны (МО) США. В 1984 г. МО США осознало, что компьютерная технология, поддерживающая распределенные данные и обмен информацией, является важной стратегией и для промышленности. Она должна помочь осуществлять все более сложные государственные закупки и позволит оснастить армию более надежными вооружением.

МО рассчитывает так же снизить время на разработку сложных вооружений и ограничить стоимость поддержки изделия в эксплуатации, которая может длиться 30–40 и более лет. При этом затраты на эксплуатацию могут намного превосходить затраты на закупку изделия.

Формально программа CALS в США началась в 1988 г. Она предполагает, что обмен технической информацией между государственными службами, поставщиками и субподрядчиками будет осуществляться электронным способом на всем протяжении жизненного цикла изделий.

В Великобритании эта инициатива началась в МО, которое в 1990 г. разработало свою собственную стратегию применения CALS, названную CIRPLS (Компьютерная интеграция требований, закупок и логистической поддержки). Была сформирована Регламентная группа, которая занялась вопросами обработки данных, стандартами, IPR (правами интеллектуальной собственности) и безопасностью, управлением данными, заключением электронных контрактов, потоками электронной информации, изменениями в составе персонала и процессах – всем, что необходимо для получения преимуществ от новой технологии. С тех пор многое удалось сделать. Поскольку основная концепция не имеет чисто военной специфики, то этот подход распространился и на промышленность: он стал применяться и при производстве товаров широкого потребления, и самолетов, и на химических предприятиях, и в компаниях по строительству дорог. Не все называют это CALS, но все могут пользоваться преимуществами этого подхода.

Все, кто нуждается в распределении и обмене информацией на протяжении жизненного цикла изделий, могут пользоваться этой технологией с пользой для себя.

Многие компании уже получили преимущества, объединив людей, процессы и технологии. Это неплохая позиция для старта в будущее. Если вы отправились в путь к новым целям, то вы обнаружите, что вы не одни и в хорошей компании, поскольку ваши конкуренты не хотят стоять на месте и тоже стартовали.

CALS интернационален и применим как в частном, так и в государственном секторе. Многие мелкие компании часто перестраиваются и извлекают выгоду быстрее, поскольку им легче перестраивать и меньше менять. Многие важные проекты уже используют CALS для снижения стоимости, затрат времени и для достижения поставленных целей с меньшими усилиями.

Мир меняется – сегодня мы можем снабжать потребителей лучшими продуктами, быстрее, дешевле и эффективнее. Мы можем делать это, работая вместе, при этом налицо экономический эффект мультиплицирования, и выигрывает вся экономика.

Делает ли это бизнес более интеллектуальным? Да, делает. Сейчас во всем мире понимают, что CALS – это средство достижения и поддержания вашей способности соревноваться с конкурентами. И если вы не воспользуетесь этим средством, то проиграете соревнование.

Следующие причины могут побудить компанию выбрать эту технологию:

- удовлетворить запросы потребителей;
- оптимизировать свои планы и программы;
- снизить стоимость, затраты времени и достичь новых целей;
- эффективно выполнять операции, не оглядываясь на границы и расстояния;
- поддержать свою конкурентоспособность;
- обрести более эффективное партнерство.

Распространение CALS. Родившись в США, эта концепция распространилась повсюду. Значительных успехов в ее освоении сегодня достигли Япония и Швеция, и многие другие страны проявляют к ней большой интерес. Проводятся международные конференции и выставки, такие как CALS Expo International, CALS Europe и UK APLS, которые хорошо посещаются и служат средством обмена опытом и достижениями.

Основные элементы технологии. Люди. CALS связан с использованием распределенной информации и совместной работой в условиях существенно более тесного сотрудничества, чем раньше. В совместной работе могут участвовать как люди, работающие в разных подразделениях одной организации, так и сотрудники разных организаций, которые пуждаются в совместной работе и использовании распределенной в разных базах данных информации о предшествующих фазах жизненного цикла изделий. Во всех случаях эти люди могут завязывать отношения в рамках так называемых виртуальных предприятий.

Есть несколько элементов, позволяющих им это делать.

Данные, данные повсюду. Ключевой идеей концепции является «создать данные один раз, использовать много раз». Это означает, что должен быть один источник цифровых данных, к которому все должны обращаться, по мере необходимости. Часто данные находятся в разных географических точках и в различных автоматизированных системах. Это означает, что техническая и деловая информация должна собираться из баз данных множества организаций-поставщиков, подрядчиков, субподрядчиков. Любое лицо, уполномоченное на это, может получить и использовать необходимую ему информацию, объем которой возрастает по мере продвижения к концу жизненного цикла изделия.

Существенную часть информации, хранящейся в базах данных, составляют данные о промышленной продукции. Обычно эта информация формируется и управляется на основе процесса, называемого Управлением данными о продукции (PDM) и включает данные, относящиеся ко всем этапам жизненного цикла изделий – разработке, прототипу, производству, техническому обслуживанию и утилизации.

Эта информация обновляется по мере необходимости. Многие организации объединяют систему PDM с системой Планирования ресурсов предприятия (ERP), которая содержит финансовую, управленческую и иную деловую информацию.

Разумное соглашение. Работа с распределенной информацией нуждается в соглашениях. Один из путей связан с созданием «Объединенной службы технической информации подрядчиков» – (CITIS). Это обеспечивает создание естественных услуг, которые создают всеохватывающую, управляемую среду распределенных цифровых данных, что улучшает деловые процессы и делает ненужным бумажный документооборот.

Первые шаги. Объединение PDM с системами автоматизированного проектирования/производства (CAD/CAM) позволяет организовать хранение всей информации об изделии. Даже когда изделие остается пока лишь неясным образом в мозгу конструктора, вы уже можете сформировать и сохранить информацию о нем и наращивать объем данных на каждой последующей стадии его жизненного цикла. Это сохранение и наращивание информации позволит вам прогнозировать будущие возможности вашего изделия – каким оно окажется в производстве, в эксплуатации, как его испытывать и обеспечивать ему поддержку в работе. Информация, содержащаяся в таких базах данных, может иметь множество форм, включая CAD модели, чертежи, отчеты, технические условия.

Вы видите на экране компьютера то, что в последствии будете иметь как изделие. Системы автоматизированного проектирования/производства предоставляют очень существенную выгоду, особенно новые 3D-системы с трехмерной графикой. Распределение процесса проектирования вовлекает в него новых исполнителей и позволяет полнее учесть требования потребителей изделий. Это особенно важно, когда изделие и его части разрабатываются и производятся в разных географических точках, поставляются разными поставщиками и должны монтироваться в единую сборку где бы то ни было.

Соединяя их вместе, электронная коммерция позволяет бизнесменам вести дела, пользуясь электронными коммуникациями и компьютерными системами, используя правила Обмена электронными данными (EDI), электронную почту e-mail, и целый набор других систем.

Параллельное проектирование использует методы «right-first-time» и параллельную работу для создания лучших, более дешевых продуктов и с меньшей затратой времени. Сроки разработки и производства – и, следовательно, сроки выхода на рынок – укорачиваются, поскольку многие работы делаются параллельно, а не последовательно, как раньше.

Устранение барьеров в проектировании, закупках и управлении вовлекает в этот процесс организации, осуществляющие поддержку

изделий в эксплуатации. Основным моментом здесь является создание Объединенной Проектной Группы (ИРТ), в которую входят потребители, менеджеры, инженеры, конструкторы, финансисты и все другие лица и организации, причастные к выполнению контракта.

Паутина, охватившая мир. Одним из наиболее значительных событий мипувших лет стало создание Интернета – событие, которое еще недавно казалось невозможным. Особое значение оно приобрело для малых и средних фирм, предоставив им средство обмена жизненно важной коммерческой информацией и документами, что сделало осуществимой и доступной Электронную Коммерцию для всех.

Преимущества CALS:

- услуги и изделия с конкурентными ценами;
- сокращение затрат времени;
- снижение общей стоимости жизненного цикла;
- снижение затрат на поддержку эксплуатации;
- повышение эффективности операций;
- точную информацию в точное время;
- улучшение взаимодействия потребителя с поставщиком;
- снижение незавершенного производства;
- возможность принимать лучшие решения;
- возможность сформировать динамичную команду, призванную сопровождать изделие на всем цикле его жизни;
- повышение квалификации персонала.

Все компании хотят производить изделия высокого качества, с конкурентными ценами и услугами. Они хотят обеспечить эффективную и экономичную поддержку своим изделиям, что привлечет и удержит покупателей. Электронная интеграция делает это возможным.

Например, сокращение стоимости и времени на стадии проектирования, производства и на всем жизненном цикле продукции. Всюду можно найти большие резервы экономии, особенно в обслуживании. Скажем, реализовать сокращение срока создания изделия до 50%. Есть примеры очень значительного снижения сроков производства, с одновременным улучшением качества, с меньшей потребностью в переделках и со снижением стоимости. Одновременно снижаются потребность в складских площадях и потери в производстве. Все это может быть достигнуто за счет использования:

а) более эффективных способов работы;

б) информационных технологий в поддержку создания, распределения и обмена сложной технической информацией между конструкторами, субподрядчиками, эксплуатационниками и т.д.

... Я должен управлять проектом. Вам придется много думать, как это все использовать. Можно серьезно выиграть – во времени и в стоимости. Для любого управленца сокращение затрат времени,

повышение качества, снижение стоимости – звучит, как музыка. Вопрос «Должен ли я это делать?» больше не стоит. Его сменил вопрос «Как я могу это делать?».

Проектная группа, объединяющая поставщиков и заказчиков, выбивает из-под ног почву для конфликтов в последствии. Члены такой группы способны работать вместе во имя общей цели.

Спихая бумажный документооборот, мы выигрываем во времени и в стоимости: меньше потребностей в командировках, проще общаться и принимать более удачные решения.

Как это сказывается на стоимости? Здесь все оказывает влияние: от какого состояния вы начали, размер вашего предприятия и к чему вы хотите прийти. И основными вопросами являются затраты и полученный выигрыш.

Если вы транснациональный изготовитель и поставщик сложного промышленного оборудования, ваши инвестиции могут исчисляться сотнями миллионов долларов. Есть несколько примеров организаций, которые инвестировали в таких объемах. Основные затраты идут на новое аппаратное и программное обеспечение, средства обучения и связи, на притирку объединенной команды проекта.

Но это не обязательно для всех. Вы можете начать со значительно более скромных инвестиций. Некоторые компании стартуют, имея в распоряжении модем, факс, мобильный компьютер и несколько мобильных телефонов. Этого достаточно, чтобы выйти на правильный путь. Не так важна цена, которую вы заплатите, но если вы не примете вызов времени, этой ценой может стать ваш бизнес.

Компьютерная среда будущего. После того как вы решите вопросы безопасности и протоколов обмена данными, ваш штаб будет готов к работе, технология и программное обеспечение окажутся в вашем распоряжении и вы зададитесь вопросом – каков же он, этот новый мир, который передо мной откроется?

Наиболее заметным последствием вашего нового стиля работы окажется то, что для вас все станет совершаться быстрее. Разработка и изготовление продукции, принятие решений, модификация изделий. Информация о продукции распределена в тех местах, где она создана, а вы беспрепятственно обращаетесь к ней, несмотря на расстояния и организационные барьеры. Ваши проблемы решаются быстрее.

Некоторые обнаруживают себя в составе «виртуальных команд», собранных со всего мира для решения определенных задач. Некоторые из этих команд существуют только на начальной стадии проекта, другие на всем его протяжении.

Работа в режиме CALS:

- электронное взаимодействие между всеми участниками;
- оптимизация затрат на жизненный цикл изделия на этапе проектирования;

- создание / ведение баз данных по промышленным изделиям;
- информация создается в цифровом виде один раз, используется многократно;
- определяются навыки / ресурсы для каждой фазы проекта;
- непрерывное обучение и тренировка команды;
- общий подход к управлению звеньями снабжения, закупкой и поддержкой жизненного цикла;
- оптимизация проекта по критериям стоимости и продолжительности жизненного цикла.

Кто получает от этого выигрыш? Люди. Возросшие квалификация и культура команды позволяют получать выгоду и потребителю, и поставщику от использования общих знаний и опыта. Непрерывное обучение персонала, тесное взаимодействие людей из разных организаций позволяют паразитировать и заимствовать профессиональные навыки.

Процесс работы. Все организации используют единые источники информации и участвуют в общей работе. Раннее вовлечение в проектирование поставщиков и потребителей делает возможным параллельное проектирование и оптимизирует поддержку жизненного цикла изделий.

Технология. Электронное распределение / обмен информацией дают существенный выигрыш во времени. Трехмерное компьютерное моделирование и Системы управления данными об изделиях предоставляют широкие возможности для улучшения проектирования и производства, снижения объема хранимых запасов полуфабрикатов. Изменения конструкции легко проводятся в электронной форме. Поддержка изделий в эксплуатации становится дешевле и эффективнее.

Как это ощущают участники процесса. Все участники проекта – потребители, заказчики, головные подрядчики и субподрядчики создают базы данных о промышленных изделиях, которые пополняют данными о конкретном изделии, пока не завершится его жизненный цикл (иногда он длится 30–40 лет).

Каждый из участников проекта ощущает выгоду по-своему:

Заказчик проекта испытывает больше доверия к тому, что происходит, поскольку непосредственно участвует в общей команде и получает информацию в режиме реального времени.

Потребитель и персонал поддержки не будут чувствовать себя отрезанными от того, что происходит у разработчика и изготовителя, поскольку получают доступ к любой информации, которая их интересует. Электронная обратная связь позволяет и разработчику лучше представлять проблемы эксплуатации и совершенствовать конструкцию.

Головные подрядчики управляют созданием изделия и взаимодействуют с подрядчиками / поставщиками в интерактивном режиме,

используя распределенную информацию для формирования документов и конструирования. Расширенный объем данных создаст условия для сокращения времени и стоимости.

Субподрядчики/поставщики являются составной частью общей команды, выполняющей проект, взаимодействуют через компьютерную сеть с головным подрядчиком в интерактивном режиме и имеют прямой доступ к системе Управления данными об изделиях (PDM). Это расширяет их возможности и делает работу более продуктивной.

Таким образом, реализация CALS-программы должна обеспечить:

- упрощение делового сотрудничества в рамках транснациональных (расширенных) предприятий вне зависимости от деятельности конкретной компании за счет применения международных стандартов. Это в свою очередь должно повысить гибкость бизнес-процессов и в то же время гарантировать максимальные шансы для выхода на рынок;
- создание модульной структуры стандартов с согласованными уровнями специализации и соответствующими определениями совместимости, которые могут быть реализованы по частям, что может быть вполне оправданным внутри отдельных предприятий (их подразделений) за счет Интегрированного подхода. При этом должна сохраняться интероперабельность элементов всей системы;
- формирование четко определенных задач по разработке и приятию CALS-стандартов с тем, чтобы промышленные предприятия могли с уверенностью планировать их внедрение. Это особенно важно для поддержки экспериментальных и производственных проектов;
- концентрирование сравнительно небольших ресурсов на наиболее эффективных направлениях разработки стандартов и проектов их реализации путем устранения необходимости совершенствования / дублирования работ;
- создание основы для гарантии конкуренции и поддержки со стороны рынка;
- устранение противоречивых стандартов и экспериментальных проектов, что позволит избежать не согласующихся друг с другом рекомендаций;
- обеспечение компаниям, реализующим результаты данной Программы, получения выгоды от расширения доступа на мировой рынок, на котором уже используются международные стандарты, создавая тем самым условия для справедливой конкуренции;
- особая роль в программе должна быть отведена подготовке и переподготовке научных и инженерных кадров.

1.1. Основные концепции CALS-технологий

Современное производство сложных изделий машиностроения подразумевает согласованную работу многих предприятий. Для обеспечения согласованной работы всех предприятий, участвующих в проектировании, производстве, реализации и эксплуатации изделий, используется соответствующая информационная поддержка этапов жизненного цикла изделий. Такая поддержка и компьютерное сопровождение жизненного цикла изделия получили название CALS (Continuous Acquisition and Lifecycle Support). Назначение CALS-технологий – обеспечивать представление необходимой информации в нужное время, в нужном виде, в конкретном месте любому пользователю на всех этапах жизненного цикла изделия. Внедрение CALS-технологий в ближайшие годы станет необходимым условием выживания промышленных предприятий при существующей жесткой конкуренции товаров на международных и национальных рынках.

В современных условиях участниками жизненного цикла конкретного изделия могут быть юридически и территориально не связанные друг с другом предприятия. CALS-технологии призваны служить средством, интегрирующим существующие на предприятиях автоматизированные системы обработки информации в единую функциональную систему. Главная задача создания и внедрения CALS-технологий – обеспечение единообразных описаний и смысловой интерпретации данных независимо от места и времени их получения в общей системе, имеющей масштабы вплоть до глобальных. CALS-технологии не отвергают существующие автоматизированные

системы обработки информации (САПР, АСТПП, АСУ, АСУП и др.), а служат средством их интеграции и эффективного взаимодействия. При этом структура проектной, технологической и эксплуатационной документации, понятийный аппарат и языки представления данных должны быть стандартизованы.

По аналогии с системами автоматизированного проектирования в составе CALS различают лингвистическое, информационное, математическое, программное, методическое, техническое и организационное обеспечение системы.

К *лингвистическому обеспечению* CALS относятся языки и форматы данных о промышленных изделиях и процессах, используемые для представления и обмена информацией на всех этапах жизненного цикла изделий.

Информационное обеспечение составляют базы данных, содержащие сведения о промышленных изделиях, используемые различными системами в процессе проектирования, производства, эксплуатации и утилизации изделий. В состав информационного обеспечения входят также серии международных и национальных CALS-стандартов и спецификаций.

Математическое обеспечение CALS включает, модели и алгоритмы взаимодействия различных систем и их компонентов в CALS-технологиях. К этим моделям относятся методы структурного и имитационного моделирования, методы планирования и управления процессами, распределения ресурсов и т.п.

Программное обеспечение CALS представлено программными комплексами, предназначенными для поддержки единого информационного пространства на всех этапах жизненного цикла изделий. Это системы управления документами и документооборотом, управления проектными данными, взаимодействия предприятий в электронном бизнесе, подготовки интерактивных электронных технических руководств и некоторые другие.

Методическое обеспечение CALS представлено методиками выполнения таких процессов, как структурирование сложных объектов, их функциональное и информационное моделирование, параллельное (совмещенное) проектирование и производство, объектно-ориентированное проектирование, создание онтологии приложений.

К *техническому обеспечению* CALS относят аппаратные средства получения, хранения, обработки и визуализации данных при информационном сопровождении изделий. Взаимодействие частей виртуальных предприятий, систем, поддерживающих разные этапы жизненного цикла изделий, происходит через линии передачи данных и сетевое коммутирующее оборудование.

- SCM – Supply Chain Management (управление цепочками поставок);
- CRM – Customer Relationship Management (управление взаимоотношениями с заказчиками);
- SCADA – Supervisory Control And Data Acquisition (диспетчерское управление производственными процессами);
- CNC – Computer Numerical Control (компьютерное числовое управление);
- S&SM – Sales and Service Management (управление продажами и обслуживанием);
- CPC – Collaborative Product Commerce (совместный электронный бизнес).

Перечисленные автоматизированные системы могут работать автономно, и в настоящее время так обычно и происходит. Однако эффективность автоматизации будет заметно выше, если данные генерируемые в одной из систем, будут доступны в других системах, поскольку принимаемые в них решения станут более обоснованными.

Чтобы достичь должного уровня взаимодействия промышленных автоматизированных систем, требуется создание единого информационного пространства не только на отдельных предприятиях, но и, что более важно, в рамках объединения предприятий. Единое информационное пространство обеспечивается благодаря унификации как формы, так и содержания информации о конкретных изделиях на различных этапах их жизненного цикла.

Унификация формы достигается использованием стандартных форматов и языков представления информации в межпрограммных обменах и при документировании.

Унификация содержания, понимаемая как однозначная правильная интерпретация данных о конкретном изделии на всех этапах его жизненного цикла, обеспечивается разработкой онтологий (мета-описаний) приложений, закрепляемых в прикладных CALS-протоколах.

Одной из важнейших проблем CALS-технологий является обеспечение единого смыслового содержания данных при использовании их на различных этапах жизненного цикла изделия. Классические подходы к реализации СУБД имеют весьма ограниченные семантические возможности. Они очень мало говорят о реальном смысловом значении данных и семантических требованиях к ним. Важность этой проблемы в CALS обусловлена тем, что различные системы могут использовать одни и те же данные – данные об одних и тех же объектах – но с разных точек зрения, при этом одни и те же данные могут использоваться в одно и то же время при параллельном решении разных

задач. Поэтому в CALS технологиях широко используются концептуальные модели данных, которые в большей степени отражают смысловое содержание информации.

1.2. Научные проблемы CALS-технологий

В предыдущем разделе главы и во введении были обрисованы области существования и области действия CALS-технологий. В данном параграфе установим связь между областями CALS-технологий и возникающими при этом научными проблемами (см. табл.1).

Таблица 1

| Области CALS-T | Научные проблемы |
|---|---|
| Методы анализа и реинжиниринга бизнес-процессов | Методология создания оптимальной технологической среды ¹⁾ , включающая вопросы новейших технологий машино- и приборостроения, организационно-экономического управления, конструкторско-технологического проектирования и соответствующего информационного обеспечения средствами вычислительной техники и передачи данных по сетям |
| Методы и средства параллельного проектирования | Методология разработки сложных изделий путем их декомпозиции и параллельного ведения отдельных проектов |
| Технология логистики | Теория управления сложными системами организационно-экономического и организационно-технологического типа. |
| Технологии Интернет/ Интранет | Теория передачи информации по телекоммуникационным компьютерным сетям |
| Унифицированная модель изделия (STEP) | Теория создания многоаспектных математических моделей изделий (функциональных, геометрических (конструкторских), технологических и т.п.) |
| Электронная документация на изделие | |

1.3. Интегрированные производственные системы

Определим подход к общему понятию производственной системы, отражающий идею структурно-функциональной декомпозиции – основного метода анализа сложных организационно-технических систем. В его основе лежит понятие так называемой открытой системы, получившее особое распространение в связи с работами по архитектуре сетей ЭВМ, трактуемых как взаимодействие открытых систем.

1) Под технологической средой понимается совокупность технологических процессов, а также методов и средств их проектирования, и реализующая их технологическое оборудование (станки, инструменты, системы управления и контроля и т.п.), предназначенная для выпуска изделий, потребных рынку.

Одним из результатов исследований в этом направлении явилось ясное представление о замкнутой на рынок модели расширенной производственной системы, которую схематически можно представить следующим образом (рис.1.2.):



Рис.1.2. Модель расширенной производственной системы, замкнутой на рынок

В реальных условиях схема удовлетворения потребностей сложнее и может включать в себя, кроме производственного цикла, циклы распределения, потребления и утилизации отходов. Совокупность всех этих циклов образует производительную сеть для удовлетворения рассматриваемой потребности рынка. Рассматриваемые задачи и составляют содержательное ядро компьютеризованных информационных систем, охватывающих все этапы жизненного цикла изделия. Информационные системы такого типа получили за рубежом название CALS (Continuous Acquisition and Life Cycle Support – поддержка жизненного цикла изделия), а процессы их функционирования – CALS-технологий.

Как уже было сказано, целью CALS-технологий является разработка методологии ускорения ведения проектов по созданию

объектов высоких технологий, т.е. увеличение производительности труда на всех этапах жизненного цикла создания изделия.

Детальная разработка понятия открытой производственной системы является задачей дальнейших исследований. Хотелось бы надеяться, что их результатом станет архитектура интеграции открытых производственных систем (OMSI – Open Manufacturing System Integration), создающая общую концептуальную и терминологическую базу для работ по автоматизации производства и CALS-технологиям.

Понятия изделия и его жизненного цикла являются центральными в рассматриваемой концепции производственной системы (ПС). Они лежат в основе анализа организационной структуры ПС, а также методологии создания новых видов сложной продукции. Более того, рассматривая саму ПС как сложное изделие, можно в рамках общего подхода исследовать задачу построения методологии проектирования, внедрения и сопровождения производственных систем.

Обобщая, можно сказать, что на повестку дня выходит проблема разработки общей концепции управления созданием изделий, в значительной степени независимой от их конкретного вида.

Под изделием мы будем понимать сложный технический объект, удовлетворяющий долговременные интересы многих потребителей и допускающий, вообще говоря, множество вариантов его использования самими потребителями без непосредственного контакта с производителем.

Данное определение подчеркивает главные черты изделия:

- наличие потребительской стоимости (служебное назначение);
- сложность проектирования и/или изготовления;
- наличие многих (и даже непредвиденных) вариантов использования;
- возможность использования без непосредственной помощи изготовителя;
- долговременный характер использования (несколько лет и более).

Примерами изделий в смысле данного определения могут служить такие виды промышленной продукции, как станки, автомобили. ЭВМ, телевизоры, программное обеспечение.

В настоящее время все большее распространение получает концепция управления предприятием на основе понятия жизненного цикла изделия, под которым понимают интервал времени от момента осознания потребности в создании изделия до момента окончания его обслуживания у пользователя. Жизненный цикл изделия имеет определенную структуру, т.е. он разбивается на ряд отрезков. Для каждого вида изделий могут существовать свои варианты разбиения жизненного цикла.

Здесь мы рассмотрим один из таких вариантов, который может быть в дальнейшем уточнен. Жизненный цикл разбивается на ряд фаз. В каждой фазе решаются свои специфические задачи в общем процессе создания изделий. Фазы могут частично перекрываться, что соответствует естественному ходу вещей и, кроме того, позволяет сократить сроки создания изделия.

Предлагаемый вариант декомпозиции включает в себя следующие фазы: анализ потребности в изделии, проектирование изделия, техническая подготовка производства изделия, производство изделия, продвижение изделия к потребителю, использование и утилизация изделия.

Разбиение жизненного цикла на фазы позволяет произвести в первом приближении декомпозицию задачи управления жизненным циклом изделия.

Однако ее недостаточно для определения функциональной структуры предприятия, создающего новые изделия. Необходима дополнительная декомпозиция по функциям управления жизненным циклом.

В число функций управления жизненным циклом входят: менеджмент, обеспечение качества, разработка, изготовление, сопровождение, документирование, обслуживание.

Каждая функция имеет свою интерпретацию для каждой фазы. Так, например, функция изготовления в фазе технической подготовки может включать в себя изготовление опытного образца, а в фазе производства выпуск серийных (промышленных) образцов. Соответственно, в этих двух фазах существенно различаются требования к уровню автоматизации, методы планирования и др.

В результате получается матрица задач «фаза–функция», которая фактически определяет матричную структуру предприятия. (рис.1.3).

На втором уровне декомпозиции можно рассматривать разбиение фаз на более мелкие этапы и/или уточнить список функций. В результате получается новый набор матриц, каждая из которых соответствует клетке исходной матрицы, определяющей декомпозицию первого уровня.

Подобная процедура декомпозиции, следующая методологии структурного анализа, носит регулярный характер, что позволяет упорядочить процесс построения организационной структуры предприятия и структурировать сам процесс управления его (этого предприятия) жизненным циклом.

Из сказанного следует, что управление производственной системой, по существу, означает параллельное управление жизненным циклом изделий, одновременно существующих в системе.

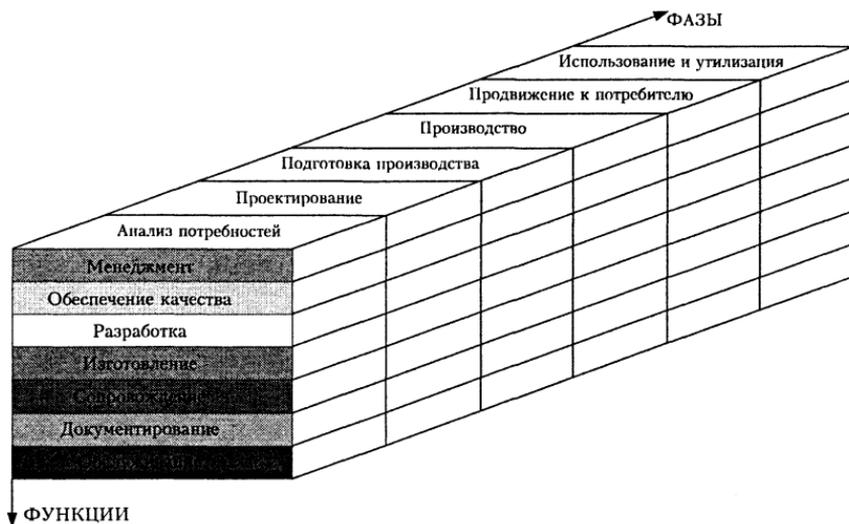


Рис.1.3. Функции управления Жизненным Циклом Изделия

Введение понятий изделия и функций управления процессом его создания позволяет уточнить понятие интегрированной производственной системы.

А именно под интегрированной производственной системой (ИПС) естественно понимать производственную систему, которая:

- ориентирована на выпуск именно изделий;
- поддерживает полный жизненный цикл выпускаемых изделий от момента осознания потребности в новом изделии до сопровождения его у потребителя.

В рамках предлагаемого подхода ИПС выступает как теоретическая модель компьютеризированного интегрированного производства.

Фактическое осуществление интеграции предполагает решение следующих вопросов:

- декомпозиция задач (функций) управления жизненным циклом изделия;
- разработка организационной структуры и схемы взаимодействия подразделений;
- построение конкретных методов решения частных задач (т.е. осуществления функций) управления жизненным циклом и их реализация.

Ранее был рассмотрен регулярный процесс декомпозиции задач на основе матрицы «фаза-функция». Соответственно, задача определения структуры предприятия может трактоваться как задача

построения отображения матрицы задач на структуру подразделений при следующих ограничениях:

- сохранение функциональности подразделений;
- минимизация связей между подразделениями;
- сохранение размеров подразделений (на каждом уровне иерархии) в разумных пределах для обеспечения управляемости;
- обеспечение гибкости реорганизации.

Логика взаимодействия подразделений определяется последовательностью фаз жизненного цикла, а также связями между частными задачами управления внутри каждой фазы.

Можно сформулировать основные требования к автоматизации производственных систем:

- автоматизация на каждом этапе должна охватывать функционально законченную в организационном, производственном и управленческом плане подсистему, с тем чтобы уменьшить трудности ее интеграции в реальный производственный процесс;
- автоматизированная система должна иметь развитую организационно-методическую поддержку разработки, внедрения, обучения, использования и сопровождения;
- автоматизированная система должна быть открытой, т.е. для нее должен быть определен на программно-аппаратном уровне интерфейс, обеспечивающий ее стыковку с другими автоматизированными системами;
- для автоматизированной системы должен быть предусмотрен регулярный механизм структурной, функциональной и параметрической настройки на конкретные условия применения;
- в рамках автоматизированной системы должен быть обеспечен удобный для персонала интерфейс, рассчитанный на неподготовленного пользователя, не являющегося программистом; общение (диалог) системы с пользователем должно вестись в терминах и понятиях предметной области, привычной для пользователя.

Введенные выше понятия образуют концептуальную базу исследований по созданию компьютеризированных интегрированных производств. Следующим шагом является разработка их архитектуры.

Под архитектурой технической системы (в данном случае ИПС) понимается структурно-функциональная модель, описывающая:

- состав системы (элементы: подсистемы, уровни, компоненты);
- функции элементов;
- связи и взаимодействие (интерфейсы) элементов;
- правила композиции элементов.

Важной особенностью архитектуры является то, что каждая выделяемая при декомпозиции подсистема имеет вполне определенное функциональное назначение. Более того, уточнение функции подсистемы достигается с помощью описания ее интерфейса, т.е. полного набора ее связей (материальных, информационных, алгоритмических и т.д.) с объемлющей системой, а точнее с другими системами. Фактически, описание интерфейса подсистемы определяет ее внешнее поведение, в то время как внутренние механизмы реализации этого поведения не рассматриваются (на определенном уровне детализации).

Таким образом, архитектура определяет модель множества реальных систем, имеющих различную реализацию, но эквивалентных в том смысле, что они построены из одного набора компонентов по одинаковым правилам, полностью определяемым набором описаний интерфейсов компонентов.

Подобный подход к проектированию систем, основанный на функциональной структуризации, обладает следующими основными преимуществами:

- особое внимание уделяется функциональному описанию системы, которое является наиболее важным и трудоемким при проектировании систем со сложным поведением;
- каждая функция выделяется как самостоятельная лишь в том случае, когда возможно ее определение через четкое и неоднозначное описание внешнего поведения компонента, реализующего данную функцию; данное условие обеспечивает строгость описания системы на каждом уровне детализации, способствуя в конечном счете улучшению качества проекта;
- определяются предпочтительные варианты реализации системы, в которых граница между «реальными» подсистемами проходит по границе между подсистемами, выделенными в структурно-функциональной модели; таким образом, реализация сохраняет логическую структуру, что повышает ее «понятность» и облегчает в дальнейшем обучение и сопровождение;
- обеспечивается возможность распараллеливания на возможно более ранней стадии работ по проектированию и реализации системы;
- облегчается стыковка компонентов и комплексная отладка системы за счет наличия четких требований к взаимодействию подсистем и компонентов;
- облегчается развитие системы за счет возможности улучшения реализации, а также удаления / добавления отдельных элементов при соблюдении их системных интерфейсов; при этом изменения всегда носят «локальный» характер: изменение

внутреннего механизма функционирования отдельного компонента при сохранении его внешнего поведения не нарушает целостности системы. Примером архитектуры сложной технической системы может служить эталонная модель соединения открытых систем, описывающая принципы иерархически модульного построения сетей ЭВМ.

Из сказанного можно сделать некоторые выводы.

1. Рассмотрен ряд проблем концептуального проектирования интегрированных производственных систем и намечен подход к их решению, который может быть использован при создании методологии сквозного проектирования интегрированного автоматизированного производства – от общей концепции производственной системы до разработки программно-аппаратного обеспечения

2. Тенденции развития современного производства выдвигают задачу создания единой концепции нового поколения производственных систем, охватывающей вопросы экономики, организации, проектирования, технологии и управления. Проведенный анализ показывает, что базой для создания подобной концепции целесообразно выбрать понятия изделия, жизненного цикла изделия, функций управления жизненным циклом и открытой системы.

Предлагаемый подход позволяет:

- придать определенность понятию интегрированного производства;
- провести регулярную декомпозицию задач управления производственной системой;
- трактовать управление производственной системой как параллельное управление жизненными циклами изделий, одновременно существующих в системе;
- рассматривать производственную систему как регулярную композицию открытых систем, полностью описываемых своим внешним интерфейсом;
- применить общие решения к созданию производственных систем, не поддерживающих полный жизненный цикл, обеспечив тем самым их совместимость и упорядоченное развитие в смысле расширения состава функций управления жизненным циклом изделия и повышения уровня их автоматизации.

3. Локальная автоматизация (т.е. автоматизация отдельных функций) управления производством не может дать радикального эффекта. Поэтому необходима комплексная автоматизация на базе целостной архитектуры системы управления. Разработка подобной архитектуры в рамках общей концепции производственной системы позволяет:

- определить области и направления стандартизации в сфере комплексной автоматизации производства;
- обеспечить гарантированный уровень функциональных возможностей системы управления;
- обеспечить инвариантность системы управления по отношению к конфигурации технологического оборудования и средств управления,
- облегчить комплексирование оборудования, а также программно-аппаратных средств, имеющих стандартные в рамках архитектуры внешние интерфейсы, а следовательно, сократить затраты на индивидуальное проектирование;
- сократить время проектирования и реализации за счет распараллеливания работ по подсистемам при условии соблюдения их внешних интерфейсов,
- создать автоматизированную систему структурно-параметрической настройки (генерации) программного обеспечения на конкретные конфигурации оборудования и средств управления;
- облегчить сопровождение и модификацию системы за счет локализации вносимых изменений.

4. Анализ функциональной структуры интегрированной производственной системы показал, что она носит существенно распределенный характер – в том смысле, что каждая из функций управления жизненным циклом имеет свою интерпретацию во всех фазах жизненного цикла на всех уровнях организационной иерархии. При этом подавляющее большинство связей между функциями носит организационный характер. Из этого следует, что производство следует рассматривать как коллектив распределенных систем организационного типа, обладающих своим поведением. Соответственно, требуется развитие новых методов исследования и проектирования организационно-технических систем, ориентированных прежде всего на описание и учет их поведенческого аспекта.

В качестве примера реализации указанного выше приведем Предложение фирмы Галика АГ по системам автоматизированного проектирования и подготовки производства.

Фирма ГАЛИКА АГ, Швейцария, с начала 1980-х годов занимается поставками прецизионного технологического оборудования для машиностроительных заводов, а также программно-технических комплексов для проектирования различных изделий и подготовки их производства, т.е. CAD/CAM/CAE системы и решения. Работа с клиентом предусмотрена не только в части поставки конкретных систем, но и в работе по обеспечению системной интеграции поставляемых комплексов, их соответствия существующим деловым и технологическим процессам.

Стратегия системной интеграции. Глобальная задача системной интеграции предприятия как практической деятельности состоит в том, чтобы обеспечить его структурное построение и функционирование как единого целого на основе средств информационной технологии и техники управления с целью вывести предприятие на качественно новый, высокий, уровень показателей продуктивности и эффективности. Системная интеграция предполагает целенаправленное реформирование предприятия согласно разработанной модели (корпоративной модели), которое называется реинжинирингом (более традиционно реинжинирингом бизнес-процессов) и охватывает практически все сферы деятельности предприятия.

В части системной интеграции фирма исходит из того, что бизнес-процессы на машиностроительном предприятии традиционно допускают деление на следующие категории:

- маркетинг и новые разработки;
- конструирование изделий;
- технологическая подготовка производства изделий;
- материальное снабжение;
- производственные процессы;
- сбыт;
- процессы управления различными ресурсами (финансы, производственная инфраструктура, оборудование, персонал и т.д.).

Естественно, различные процессы в рамках работы по системной интеграции подкрепляются различными программно-аппаратными средствами, примерно как показано на рис.1.4.

Примерная структура связи бизнес-процессов и программно-аппаратного комплекса фирмы Галика. Рассматривая предприятие как единое целое, в указанном спектре процессов фирма концентрирует свою деятельность только вокруг конструирования и технологической подготовки производства, рассматривая их как наиболее наукоемкий и взаимосвязанный комплекс «технических бизнес-процессов». В основе решения задачи реинжиниринга технических бизнес-процессов машиностроительного предприятия фирма использует американскую систему Optegra фирмы Computervision, т.е. задача увязывания и оптимизации технических процессов решается главным образом средствами системы Optegra.

Реинжиниринг остальных процессов, условно называемых «общими бизнес-процессами», фирма предпочитает проводить в партнерстве с другими фирмами, специализирующимися в секторе нетехнической деятельности предприятия, и которые являются поставщиками систем управления по общим бизнес-процессам (корпоративных систем). Например,

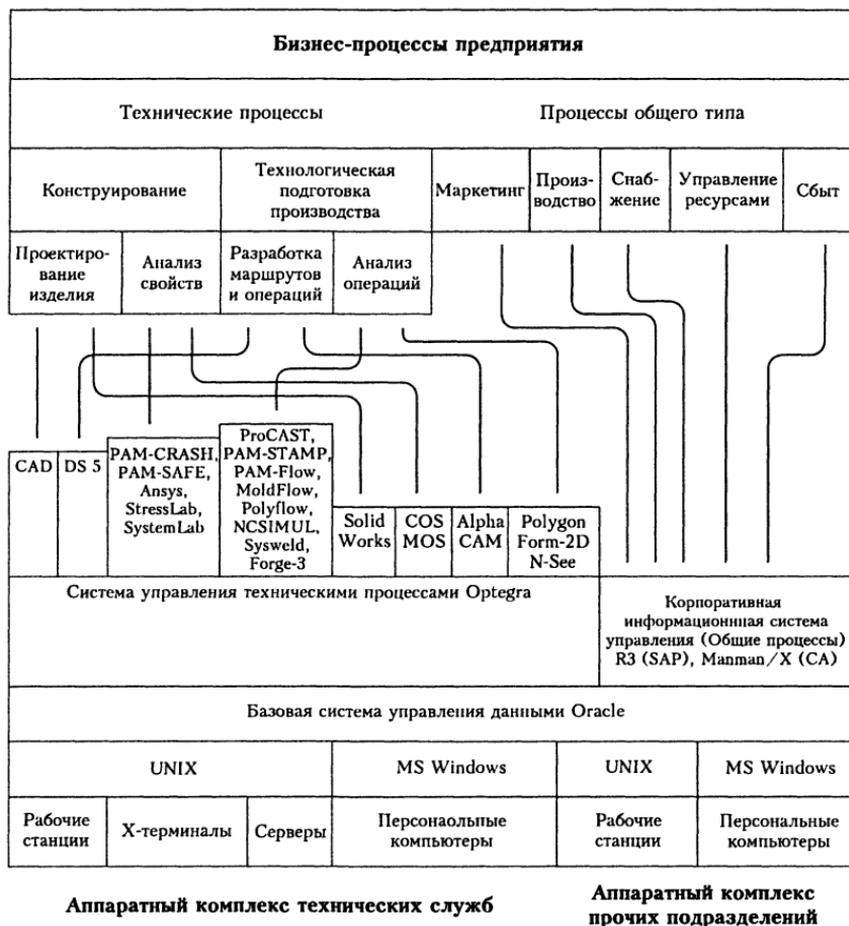


Рис.1.4. Программно-техническая поддержка бизнес-процессов предприятия

- SAP с системой R3;
- Computer Associates с семейством систем MANMAN / X. Masterpiece, Accpac...;
- SCALA с одноименной системой;
- Cintel с системой SIMPLAN и т.д.

Выбор партнера по части общих процессов в сильной степени зависит от пожеланий, от истории и традиций работы предприятия. Партнера следует выбирать до начала проектных работ, поскольку первым этапом в технологии реинжиниринга является обследование предприятия, т.е. создание его компьютерной модели, а это в сильной степени определяет технику проведения обследования.

Основные этапы работы технического реинжиниринга. Работа фирмы по реформированию технических бизнес-процессов (техническому реинжинирингу) предприятия распадается на этапы:

- обследование технических подразделений, разработка плана их реформирования;
- внедрение общей системы управления техническими процессами согласованно с проектом корпоративной системы управления предприятием;
- внедрения комплекса конкретных технических систем.

Работа по обследованию. Обследование существующей структуры основных технических подразделений предприятия и соответствующих процессов по методике структурного анализа (SADT), в результате которого формируется компьютерная модель фактических процессов. На основе процессов фактической модели («как есть») формируется модель-эталон («как надо»), после одобрения которой руководством фирма разрабатывает конкретный план организационно-технических мероприятий реформирования предприятия к переходу для работы в соответствии с эталонной моделью.

В результате проведения работ по обследованию предприятие получает:

- модель организационной структуры технических подразделений;
- функциональную и информационную модель существующих процессов (фактическую модель);
- полученную на основе фактической модель, соответствующую представленным руководством целям и требованиям (эталонную модель);
- детальный план реформирования технических подразделений для постепенного перехода от фактической к эталонной модели;
- для поддержки разработанных моделей в актуальном виде в структуре завода создается специальная постоянно работающая группа системного анализа и развития, укомплектованная компьютерными и специальными программными средствами (аналитическая группа).

Проект системы управления для технических бизнес-процессов. По материалам обследования осуществляется реинжиниринг бизнес-процессов и вырабатывается новая структура технических служб, согласуются их функции. Фирма Галика основывает свой проект на внедрении системы интеграции технических процессов Ortega. Система построена на технологии «клиент-сервер», т.е. предполагает внедрение компьютерной сети рабочих мест исполнителей с центральным оборудованием (общими серверами для данных и программ) и клиентскими рабочими местами (CAD/CAM-системами).

Optegra практически не имеет ограничений по количеству рабочих мест и интенсивности информационного обмена, географических масштабов и особенностей предприятия и реализует интенсивно разрабатываемую на Западе концепцию «сквозного проекта», или электронного определения продукта EPD (Electronic Product Definition). Суть этой концепции в том, что рабочие места исполнителей, обязательно оборудованные САД-системами (возможно разного типа, но стыкуемыми с Optegra через интерфейсы) под управлением системы Optegra, дают возможность на уровне конструирования планировать ход разработки, визуализировать его объемную модель, видеть «дерево изделия» всем участникам процесса, для которых это необходимо (менеджеры, главные специалисты, реализующие или контролирующие сборку). Результатом этой работы является полностью собранный в виртуальном объеме на рабочей станции главного конструктора электронный прототип изделия, который можно дальше подвергать компьютерному анализу и декомпозировать на узлы и детали для разработки технологических маршрутов их изготовления и сборки. Система из электронного прототипа изделия автоматически подготавливает спецификации покупных изделий для служб внешней кооперации предприятия.

На уровне технологической подготовки производства система Optegra объединяет технологов и конструкторов технологической оснастки единым сетевым планированием, позволяет формировать и отслеживать технологические маршруты, извлекая из них спецификации по необходимым материалам, инструменту, оснастке, трудозатратам для служб материального обеспечения и вспомогательных производств.

Таким образом система Optegra создаст оптимальную деловую среду для групповой параллельной работы конструкторов, технологов, менеджеров и прочих исполнителей, подчиняя их действия заранее отработанным процедурам и обеспечивая информационную стыковку с системой управления общими процессами.

Поставка конкретных технических систем. Конкретные технические системы могут поставаться в рамках глобального проекта реконструкции на основе отработанной конфигурации системы Optegra или по отдельным заказам для решения каких-либо локальных конструкторских или технологических задач. Программа поставок по линии конкретных систем логично распадается на принципиальные два уровня:

- системы высокого уровня для сложных изделий и процессов, базирующиеся на рабочих станциях по ОС UNIX и требующие существенных капитальных вложений;
- системы для изделий среднего уровня сложности, базирующиеся на персональных компьютерах под MS Windows и ориентированные на относительно скромный бюджет.

Разделение на два уровня оправдано практикой с точки зрения финансовых возможностей клиентов, а также тем обстоятельством, что всегда были и будут как сложные, так и простые конструкции и детали, которые требуется проектировать и производить. Предложение второго уровня может рассматриваться как наиболее дешевый стартовый этап предприятия для фундаментального перевооружения в дальнейшем. Естественно, оба уровня могут комбинироваться в зависимости от конкретных задач, однако внедрение технологии EPD возможно только с участием систем первого уровня.

Предложение по системам первого уровня для рабочих станций под OS UNIX. В качестве базовой системы проектирования и подготовки производства фирмой Галика предлагается американская CAD/CAM система CADDS5 фирмы Computervision, которая обладает достаточно развитым составом конструкторских модулей для проектирования изделий сложной геометрической формы и сборок, содержащих кроме традиционных механических деталей разнообразные подсистемы, такие как трубопроводы, кабельные жгуты, электронные компоненты. В области инженерного анализа (область CAE) предложение фирмы разделяется на два класса в зависимости от области приложения:

- анализ свойств объектов;
- анализ различных технологических процессов и операций по типу используемых в производстве технологических переделов.

Анализ свойств объектов предусмотрен в следующих областях:

- прочностные линейно-статические свойства деталей – система StressLab (модуль системы CADDS5);
- прочностные нелинейные статико-динамические свойства деталей – система ANSYS одноименной американской фирмы;
- кинематические свойства конструкций – система SystemsLab (модуль системы CADDS5);
- сопротивляемость конструкций и транспортных средств (автомобилей, судов, самолетов) ударным нагрузкам и разрушающим воздействиям в аварийных ситуациях – система PAM-CRASH французской фирмы PSI, которая может быть дополнена специальной системой анализа эффективности средств безопасности человека – системой PAM-SAFE;
- сопротивляемость конструкций и различных изделий вибрационным и ударным нагрузкам (например, оценка прочности хрупких электронных изделий или прочности и долговечности автомобильных шин) – система PAM-SHOCK французской фирмы PSI;

- анализ обтекаемости элементов деталей и конструкций воздушными и гидротоками (например, рабочие процессы в турбомашинах) – система PAM-FLOW французской фирмы PSI.

В части компьютерного моделирования технологических процессов предусмотрены следующие возможности:

- механообработка на станках с ЧПУ с целью верификации корректности программ обработки – система NCSIMUL французской фирмы Spring;
- литье металлов – система ProCAST американской фирмы UES и швейцарской фирмы Calcom;
- литье пластмасс – система MoldFlow одноименной австралийской фирмы;
- холодная листовая формовка (3D вытяжка) – система PAM-STAMP французской фирмы PSI;
- экструзия пластичных материалов – система PolyFlow одноименной бельгийской фирмы;
- горячая объемная штамповка в 3D – система FORGE3 французской фирмы Transvalor;
- термообработка деталей и сварка конструкций – система SYS-WELD французской фирмы Framatome.

Предложение второго уровня. В качестве базовой системы проектирования предлагается новая недорогая CAD система SolidWorks 97 Plus американской фирмы SolidWorks Corp. В части программирования обработки на станках предлагается также недорогая система AlphaCAM английской фирмы Licom, хорошо совместимая с SolidWorks 97.

Все возможности анализа свойств объектов реализуются недорогими системами американской корпорации Structural Research COSMOS/M или COSMOS/Works для решения следующих задач:

- линейные и нелинейные статико-динамические нагрузки деталей и конструкций;
- усталостные нагрузки;
- линейные и нелинейные термические процессы;
- процессы гидродинамики и обтекания;
- линейные и нелинейные электромагнитные явления;
- статические, квазистатические и частотные процессы в электронных цепях в условиях большого разброса параметров компонентов.

Наиболее перспективным представляется интегрированный пакет COSMOS/Works, работающий чисто на базовой геометрии системы SolidWorks.

Компьютерное моделирование некоторых технологических процессов реализуется в областях:

- механообработка на станках с ЧПУ – система N-See американской фирмы Microcompartibles;
- литье металлов – российская система Polygon;
- литье пластмасс – система MoldFlow/Flow-E;
- горячая объемная штамповка в 2D – отечественная система FORM-2D московской фирмы Квантор.

Поскольку в настоящее время наиболее актуальной для большинства предприятий является недорогая платформа на персональном компьютере, ниже приводится рекомендуемый комплект для наиболее типичных задач конструкторской и технологической практики, который по мнению фирмы максимально отвечает требованиям хорошей функциональности при предельно сжатых ценах.

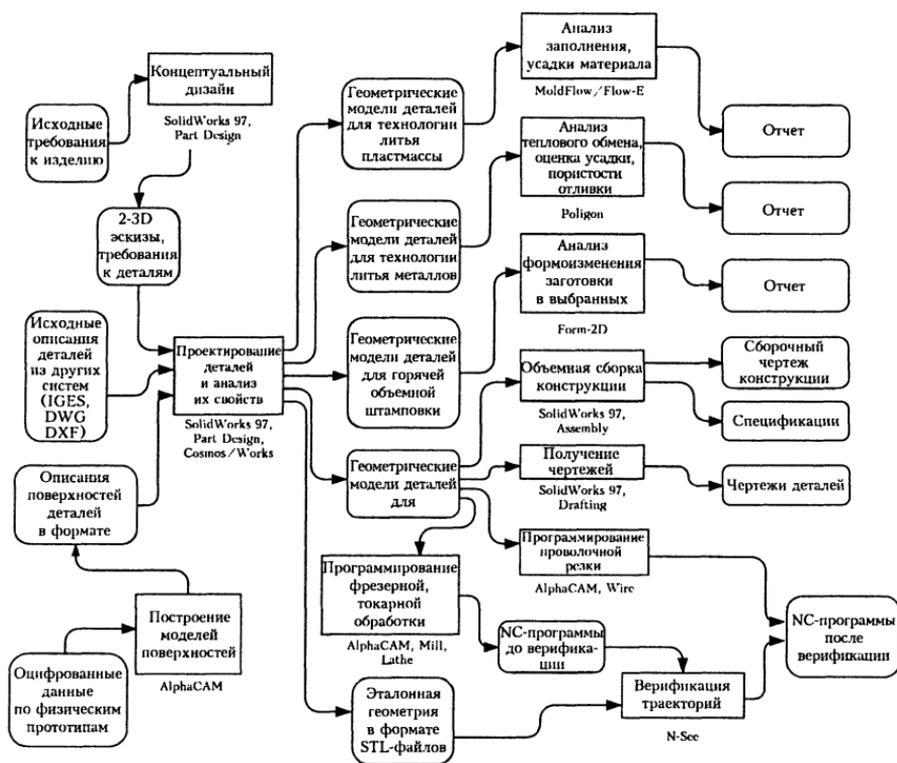


Рис.1.5. Примерная схема функциональной связи систем SolidWorks, AlphaCAM, N-See, Polygon, MoldFlow, Form-2D

Краткое описание интегрированного решения для персонального компьютера. Представленный комплект совершенно новых программных систем подобран по критерию полной интеграции по форматам данных, что иллюстрируется схемой их взаимосвязи. Полученный таким образом комплекс не претендует на изделия и детали высокой сложности, которые, как правило, реализуются в системах высокого уровня на рабочих станциях, но позволяет решать весьма широкий спектр задач по деталям и конструкциям средней и малой сложности, фигурирующих в основном в технологических подразделениях (приспособления, режущий инструмент, оснастка, типовые детали и т.д.).

Система SolidWorks 97, дополненная системой анализа COSMOS является хорошей базой для оснащения конструкторских бюро, однако высокая функциональность системы в возможностях объемной геометрии позволяет ориентировать ее как 3D систему для технолога-литейщика в комплекте с системами моделирования литейных процессов Poligon и MoldFlow/Flow-E, а также для технолога горячей объемной штамповки как базовый геометрический моделер для системы анализа формоизвлечения кузнечной заготовки FORM-2D (рис.1.5.)

На рис.1.6., 1.7. показано компьютерное сопровождение технических бизнес-процессов на платформах UNIX и Windows.

1.4. Интегрированное управление предприятием

В настоящее время имеется достаточно большое количество интегрированных систем управления предприятием как отечественных, так и зарубежных фирм-производителей. Однако наиболее продвинутой на российском и мировом рынках считается система R/3 немецкой фирмы SAP.

1.4.1. Базисная технология системы R/3 фирмы SAP

В настоящее время Базисная технология SAP является одной из наиболее часто используемых технологических платформ во всем мире для реализации решений на базе архитектуры клиент/сервер, предназначенных для решений задач хозяйственной деятельности, которые функционируют в открытых системах.

Будучи технологической основой успешно функционирующей системы R/3, Базисная технология SAP приобретает решающее значение для компании, часто становится основой архитектуры ее систем информационной технологии.

Использование отраслевых стандартов в Базисной технологии SAP также имеет важное значение и заслуживает особого внимания. При постоянной поддержке стандартов становятся возможными



Рис.1.6. Решение инженерных задач на рабочих станциях (платформа UNIX)

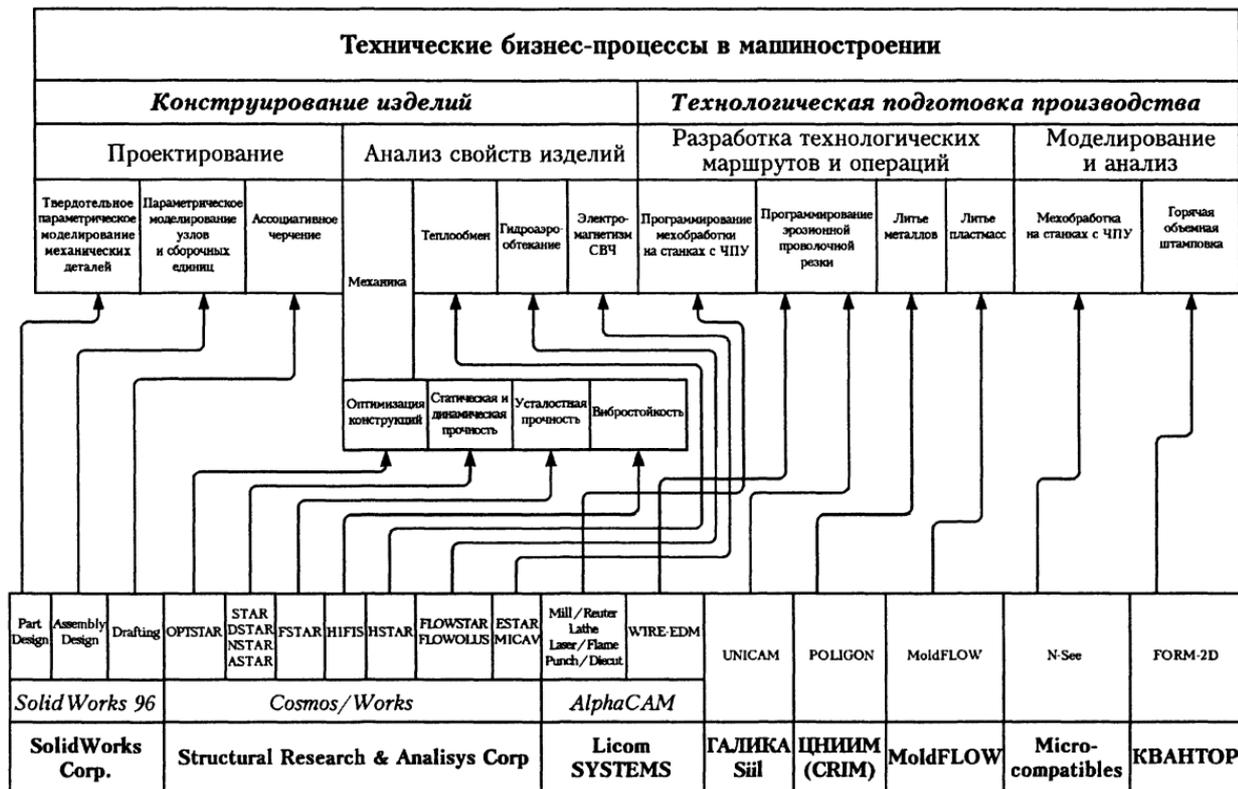


Рис.1.7. Решение инженерных задач на персональных компьютерах (платформа Windows)

«открытые» системы. Таким образом, для достижения своих целей есть возможность эффективно использовать оптимально настроенные модули.

Основное внимание уделяется этим ключевым аспектам, а также другим таким важным аспектам, как поддержка, которую предоставляет Базисная технология SAP для обеспечения распределяемости, возможности шкалирования, открытости, целостности, интегрированности, а также взаимодействия приложений на базе архитектуры клиент/сервер. При этом внимание уделяется не только обсуждению всех технических подробностей, но также и представлению общей всеобъемлющей концепции, лежащей в основе этой технологии, наряду с ее различными возможностями и полезными компонентами.

Эффективное связующее программное обеспечение. В настоящее время термин «связующее программное обеспечение» (middleware) обычно используется для обозначения программного обеспечения, которое служит связующим звеном между операционной системой и сетью (рис.1.8). В функциональном отношении оно имеет большое значение, поскольку это особое программное обеспечение абсолютно необходимо для осуществления тесного взаимодействия между всеми различными задействованными компонентами.

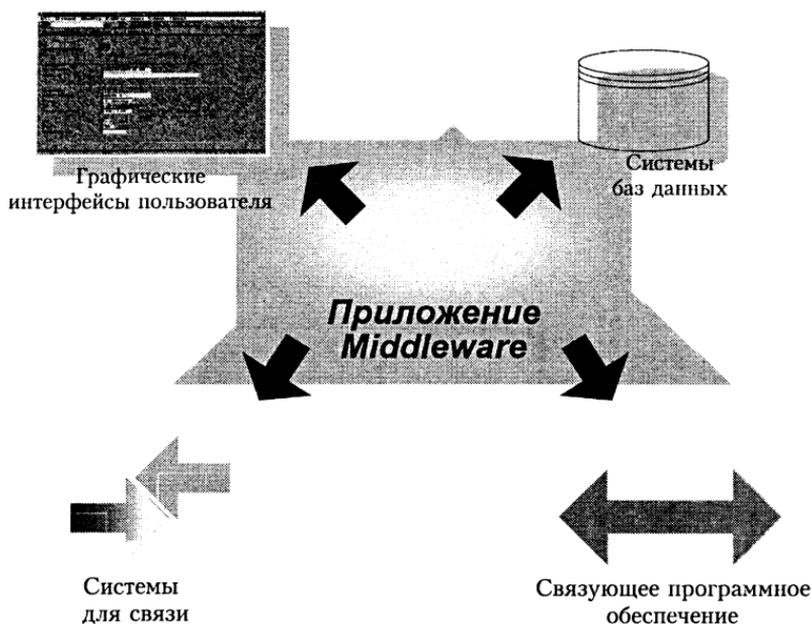


Рис.1.8. Графическая интерпретация связующего программного обеспечения

Кроме того, оно обеспечивает эффективную среду, в которой выполняются сами приложения. Поскольку имеется связующее программное обеспечение, то больше нет необходимости в контроле за различными компонентами в распределенных средах. Это позволяет пользователям и разработчикам сконцентрировать все внимание на своих фактических задачах и работать более эффективно. В результате: значительное увеличение производительности как при реализации, так и при использовании решений на базе архитектуры клиент/сервер.

Хороший способ проиллюстрировать данный подход – это провести аналогию с телефонной связью. В настоящее время очень просто набрать нужный номер по телефону в любом месте земного шара. Нет необходимости знать, какой это тип телефона (форма, модель и т.д.), а также не нужно знать все о физической природе линий телефонной связи (медный кабель, волоконная оптика, спутниковая связь и т.д.).

Да и нет необходимости знать, где находится другой телефон (возьмем, например, сотовый телефон). «Связующее программное обеспечение» отвечает за то, чтобы подключаться к нужному телефону и чтобы данные передавались и получались правильно на другом конце.

Аналогично, при обработке данных связующее программное обеспечение осуществляет тесное взаимодействие графических интерфейсов пользователя, систем управления базами данных, операционных систем и систем связи, а также обеспечивает распределяемость и возможность шкалирования приложений. Это одна из центральных задач, выполняемых Базисной технологией SAP.

Новаторские решения на основе архитектуры клиент/сервер. Новаторская Базисная технология SAP позволяет распределить приложения, работающие в среде клиент/сервер, между тремя уровнями. Сюда относится разделение функций представления данных (внешний интерфейс), прикладной логики и базы данных, тем самым достигается описанная выше распределяемость и возможность шкалирования.

Для того чтобы ответить на эти вопросы, полезно провести еще одну аналогию с телефонной связью. В настоящее время при наличии в мире миллионов телефонов просто невозможно установить прямые связи между ними. Вместо этого все задачи, связанные с данной проблемой, разбиты и распределены, таким образом достигается максимальная распределяемость, или возможность шкалирования.

Поэтому кроме телефонов, имеются также коммутационные станции и другие устройства, отвечающие за обеспечение связи, передачу вызовов и данных. Каждая такая станция отвечает в одно и то же время за большое число телефонов. Другими словами,

рассматриваемые задачи распределены. Это гарантирует возможность расширения, или, иначе говоря, возможность шкалирования телефонной связи. В любое время можно подключать новые телефоны, и при этом не будет нарушено правильное функционирование уже подключенных телефонов.

К основным целям, которые преследуются при реализации решений на основе архитектуры клиент/сервер в области обработки данных, также относится распределенная обработка и максимальная возможность шкалирования. Однако наряду с этим также важна возможность поддержки различных платформ. Подобно тому, как в настоящее время можно купить, вставить в контактное гнездо и использовать телефонный аппарат, который наиболее соответствует конкретным потребностям, в открытых средах клиент/сервер можно устанавливать самые лучшие компоненты технических средств, удовлетворяющих конкретные требования. Никто не предписывает, какие технические средства устанавливать. И задача Базисной технологии SAP состоит в том, чтобы сделать это возможным.

Распределенная обработка и возможность шкалирования. Перспективная Базисная технология SAP позволяет распределенным приложениям, функционирующим в среде клиент/сервер, удовлетворять широкий диапазон различных потребностей (рис.1.9). Все

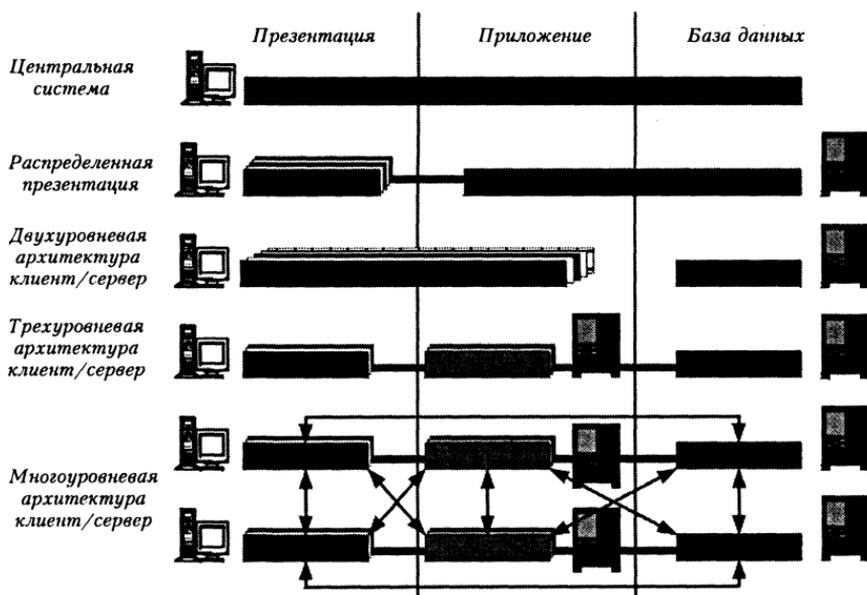


Рис.1.9. Различные архитектуры клиент/сервер

осуществимо: от использования портативного компьютера до двухуровневых систем архитектуры клиент/сервер для получения экономически эффективных вычислительных систем в малых и средних компаниях и более крупных вычислительных систем с трехуровневой архитектурой клиент/сервер для использования на очень крупных вычислительных системах, представляющих максимальную производительность и доступность системы при осуществлении обработки в трехуровневой среде клиент/сервер на параллельных серверах базы данных.

Совместная обработка в среде клиент/сервер, которая основывается на распределенных приложениях, также позволяет создавать вычислительные системы в рамках концерна или даже мирового масштаба, которые, кроме удовлетворения очень высоких ожиданий в отношении производительности и доступности системы, часто должны быть такими, чтобы их можно было приспособить к географически распределенным и уже инсталлированным приложениям существующих систем.

Естественно, что Базисная технология SAP позволяет свободно комбинировать все эти основные типы распределенной обработки. Таким образом, широкое разнообразие требований к производительности, доступности системы и защиты данных могут быть непосредственным образом удовлетворены. Технологической основой для возможности шкалирования служит поддержка, которую Базисная технология SAP обеспечивает для трехуровневой архитектуры клиент/сервер, что позволяет настраивать работу компьютера на конкретные требования и гарантирует возможность увеличения количества приложений с ростом потребностей.

Преимущества трехуровневой модели для работы в среде клиент/сервер. Нижний уровень является уровнем базы данных. Именно здесь осуществляется управление, в частности, теми данными, с которыми работает предприятие. К ним относятся как основные данные, так и данные, которые генерируются текущими процессами и записываются в память системой управления базой данных (СУБД). На уровне базы данных также находятся метаданные, управление которыми осуществляется в так называемом репозитории. Для определения данных и манипулирования ими используется отраслевой стандарт SQL (Structured Query Language – язык структурированных запросов). Он служит основой той поддержки, которую Базисная технология SAP обеспечивает в отношении ряда общераспространенных СУБД, конечно, при условии, что они отвечают требованиям, предъявляемым системой R/3 к их рабочим характеристикам. Самым важным является то, что вы можете выбрать СУБД, с которой вам хотелось бы работать. Вы продолжаете извлекать пользу от ваших

вложений в базу данных, сделанных в прошлом, и, в частности, от опыта работы, который вы накопили.

Кроме того, можно использовать свои собственные усовершенствования и расширения языка SQL для различных СУБД, и при этом пользователям не нужно иметь дело с техническими деталями. Они позволяют осуществлять непротиворечивый доступ к данным явным, прямым и эффективным способом. Такая непротиворечивая реализация общепризнанных стандартов не только поддерживает современные технологические концепции поставщиков баз данных, но также остается открытой для будущих разработок. Следовательно, вы можете извлечь пользу от нововведений без необходимости смены приложений или перехода к другой системе баз данных.

Второй уровень содержит приложения, которые настраиваются над СУБД. Приложения работают с данными, которые они извлекают из уровня базы данных и взамен вводят в него новые данные. Здесь постоянно хранятся, например, приложения системы R/3 и/или приложения, предназначенные для работы в среде клиент/сервер, которые разрабатываются, используя АВАР/4 – Инструментальные средства разработки.

Третий, самый верхний уровень – самый высокий уровень, поскольку он наиболее близок к пользователю – является уровнем презентации. Этот уровень включает в себя, в частности, интерфейс, с которым работают пользователи для доступа к приложениям, для ввода данных и просмотра результатов выполнения своих рабочих процессов.

Здесь важную роль играет поддержка графического интерфейса пользователя, позволяющая извлечь максимальную пользу из приложений, работающих в среде клиент/сервер.

Система R/3 предоставляет графический интерфейс пользователя – SAP-GUI – который оптимизирован с учетом требований существующей практики хозяйственной деятельности и создан в соответствии с последними достижениями в области эргономики.

Кроме того, благоприятное воздействие на пользователей оказывают элементы дизайна и меню, удовлетворяющие современным требованиям, подобные тем, с которыми они ознакомились, работая с общераспространенными программами для персональных компьютеров. Поэтому пользователи могут быстро изучить способы их применения и воспользоваться всеми возможностями приложений системы R/3. Базисная технология SAP позволяет использовать различные графические интерфейсы пользователя. Как правило, для каждой отдельной платформы обеспечивается одна и та же функциональность.

Следовательно, работа с приложениями в основе которых лежит Базисная технология SAP, а также то, как они выглядят, в

значительной степени не зависит от используемых платформ. Конечно, поддерживаются также особые функции данного графического интерфейса, если допускается его оптимальное встраивание с учетом концепции конкретной платформы. Кроме того, чтобы воспользоваться возможностями приложений, работающих в среде клиент/сервер, в основе которых лежит Базисная технология SAP, пользователи могут работать в знакомой среде. Пользователи очень скоро смогут работать с системой. Следовательно, во время обучения достаточно охватить только то, что является новым, что касается самих приложений. Преимущество здесь заключается в том, что обучение может проходить быстрее и с меньшими затратами.

Преимущества для пользователя за счет независимости от аппаратных средств и переносимости. Возможность распределять приложения между тремя уровнями является одним из важных аспектов Базисной технологии SAP. Но конечно, важна также поддержка конкретных аппаратных и программных средств и их взаимодействие в рамках этих уровней и между ними. Не обязательно иметь все альтернативные возможности доступными, поскольку этого нельзя сделать с требуемой тщательностью. Скорее более существенна здесь оптимальная поддержка тех альтернативных возможностей, которые имеют отношение к рынку, и, следовательно, важны для пользователей.

В частности, это означает, что можно свободно выбирать среди существующих в настоящее время операционных систем, систем управления базами данных и графических интерфейсов пользователя. Кроме того, Базисная технология SAP обеспечивает возможность легко и просто работать. Как конечным пользователям, так и разработчикам нет необходимости беспокоиться об операционной системе или СУБД. Выполнение этих задач возьмет на себя Базисная технология SAP. Пользователи могут сконцентрировать свое внимание на текущей работе, которую им необходимо сделать, и больше им не нужно терять время на решение вопросов, которые имеют отношение к самой системе.

Такие программные решения, основанные на Базисной технологии фирмы SAP, как приложения системы R/3, могут быть перенесены фактически на любую платформу. Все компоненты, поддерживаемые Базисной технологией SAP, автоматически становятся доступными для них. Следовательно, приложения могут использовать те компоненты, которые наилучшим образом подходят для выполнения их конкретных задач. Это могут быть самые последние или самые дорогостоящие аппаратные средства; это может быть самая современная технология создания аппаратных средств; или же это могут быть компоненты, уже существующие в компании.

Связь в системе. Трехуровневая модель является прямой и простой для понимания, кроме того, она упрощает существующие структуры современных архитектур клиент/сервер в информационной технологии, поскольку потоки связи существуют также в рамках каждого уровня, например между распределенными приложениями.

Таким образом, возможные операции доступа и направления, в которых они могут выполняться, должны быть соответственно переменными. Базисная технология SAP также это обеспечивает. Естественно, что для осуществления связи в сети, постоянно используются стандарты. В телефонной связи все абоненты должны соблюдать одни и те же правила (например, использовать одни и те же междугородные коды). Аналогично, вся связь в сети – неважно, локальная ли это сеть (LAN), расположенная в здании компании, или широкомасштабная, региональная сеть (WAN) – осуществляется на основе общепризнанных и соблюдаемых правил, известных как протоколы.

В качестве стандартного сетевого протокола для открытых систем установлен стандарт TCP/IP (Transmission Control Protocol/Internet Protocol). Поэтому в Базисной технологии SAP он получил развитие и стал основным протоколом для обработки в среде клиент/сервер. Все операционные системы, имеющие отношение к Базисной технологии SAP, поддерживают протокол TCP/IP, тем самым обеспечивается четкое осуществление всей связи. Интеграция с большими машинами, например, для того, чтобы осуществлять взаимодействие с системой R/2 или приложениями, разработанными третьей стороной, которые выполняются на этом компьютере, также основывается на принятом стандарте, а именно LU6.2, сетевом протоколе, который определен фирмой IBM.

Как для связи между компьютерами в рамках приложений, так и для интеграции внешних программ имеется Remote Function Call (RFC). RFC – это реализация метода Remote Procedure Call (RPC) (широко используемого в средах программирования на языке C), которую осуществила фирма SAP.

Например, приложение может использовать RFC для вызова функционального модуля системы R/2, который затем выполняется на другом компьютере. Кроме того, RFC-вызовы можно использовать для того, чтобы воспользоваться услугами внешних приложений для обработки данных. Совершенно неважно, инсталлированы ли вызываемые функциональные модули или приложения на одном или нескольких компьютерах, или осуществляется ли связь синхронно или асинхронно. А программа, вызываемая посредством RFC, может сама взять на себя инициативу и отправить свой собственный RFC-вызов. Базисная технология SAP всегда обеспечивает правильную связь между компьютерами. В результате, например, можно

воспользоваться функциями системы R/3, даже работая в других приложениях. Это исключает необходимость повтора разработок, таким образом, экономится ценное время и снижаются затраты. Кроме того, можно использовать существующую среду клиент/сервер более эффективно, поскольку функции могут выполняться на компьютерах, специально предназначенных для этой цели.

Надежность, защита и доступность системы. Пользователи ожидают, чтобы такая мощная система, как R/3 фирмы SAP, предназначенная для использования в рамках всего предприятия и играющая важную роль в его деятельности имела особенно высокую степень защиты, надежности и доступности.

Для того чтобы удовлетворить эти потребности в защите и надежности Базисная технология SAP интегрирует различные внутренние и внешние механизмы. Они охватывают системный уровень и доходят до уровня настольных компьютеров. SAP также интегрировала эффективную процедуру Kerberos, которая была разработана в известном Массачусетском технологическом институте (MIT) и впоследствии через Open Software Foundation (OSF) (Основы открытого программного обеспечения) включена в состав архитектуры Distributed Computing Environment (DCE) (Распределенная вычислительная среда). Конечно, система R/3 также включает в себя современные, тщательно разработанные функции управления доступом к данным и функциям.

Для того чтобы приложения были постоянно доступными, Базисная технология SAP предоставляет различные услуги. К ним относится Система управления вычислительным центром (CCMS), с помощью которой можно управлять всей системой и осуществлять текущий контроль этой системы на различных уровнях: уровне базы данных, прикладном уровне (например, R/3), уровне операционной системы и сетевом уровне. Система CCMS содержит процедуры и контрольные списки, позволяющие администратору системы обнаруживать потенциально критические ситуации и предотвращать их до того, как они станут острыми. Кроме того, в системе CCMS используются мониторы производительности для оптимизации системных параметров. Чтобы следить за работой всей системы архитектуры клиент/сервер, администратор системы может обратиться к графическим обзорам. Он может вмешиваться в критические области до возникновения проблем, что позволяет эффективно обеспечить доступность системы и повышает удовлетворенность пользователя ее работой.

В частности, для прикладной системы, которая подобно системе R/3 выполняет очень важную функцию в деятельности компаний, оптимальная доступность означает также сведение к минимуму времени, необходимого для перехода на новые версии. Такие новые

технологии, как Repository Switch (процедура обмена), дружественные пользователю графические интерфейсы и полностью автоматизированные процедуры инсталлирования новых версий - минимизируют время выполнения системы, поскольку больше нет необходимости в услугах дорогостоящих специалистов.

После инсталлирования новой версии необходимо также распределить программное обеспечение на отдельные серверы. В форме автоматизированного распределения программного обеспечения фирма SAP предоставляет технологию смены версий программного обеспечения без прерывания продуктивного использования системы. Для этой цели, когда бы не вызывалась транзакция, выполняется проверка для того, чтобы выяснить, содержит ли уже рассматриваемый сервер приложений новую версию. Если нет, то автоматически происходит смена версии на новую. В совокупности все эти меры позволяют работать более производительнее - долгий период непродуктивного простоя во время смены версий остался в прошлом.

Интеграция между приложениями посредством Application Link Enabling. Базисная технология SAP обеспечивает более, чем только техническую поддержку для инфраструктуры клиент/сервер. Для иллюстрации имеет смысл еще раз обратиться к упомянутому выше примеру с дистанционной связью.

Для того чтобы пользующиеся телефонной связью могли понимать друг друга, существует еще одно требование, которое должно удовлетворяться: обе стороны должны говорить на одном и том же языке. Это – единственный способ обеспечения эффективной связи. В условиях обработки данных в среде клиент/сервер это означает, что приложения могут вместе содержательно работать, если они говорят на общем языке. Базисная технология SAP для этого также предоставляет технические средства. Она позволяет различным приложениям осуществлять связь между собой – эффективно «говорить на одном и том же языке». Делая это, Базисная технология SAP выходит далеко за рамки функциональности, обычно предоставляемой «связующим программным обеспечением» (middleware). Application Link Enabling (ALE). Посредством ALE-технологии создается общий семантический уровень для различных приложений, который, таким образом, становится их общим языком (рис.1.10).

ALE допускает слабое связывание различных приложений и особенно различных систем R/3, которые могут быть инсталлированы локально, быть технически независимы друг от друга и могут даже относиться к различным версиям. Следовательно, в различных отделениях или филиалах компании можно использовать отдельные системы R/3. Для каждой субэксплуатации могут быть оптимально спроектированы стратегии внедрения и смены версий, и при этом нет

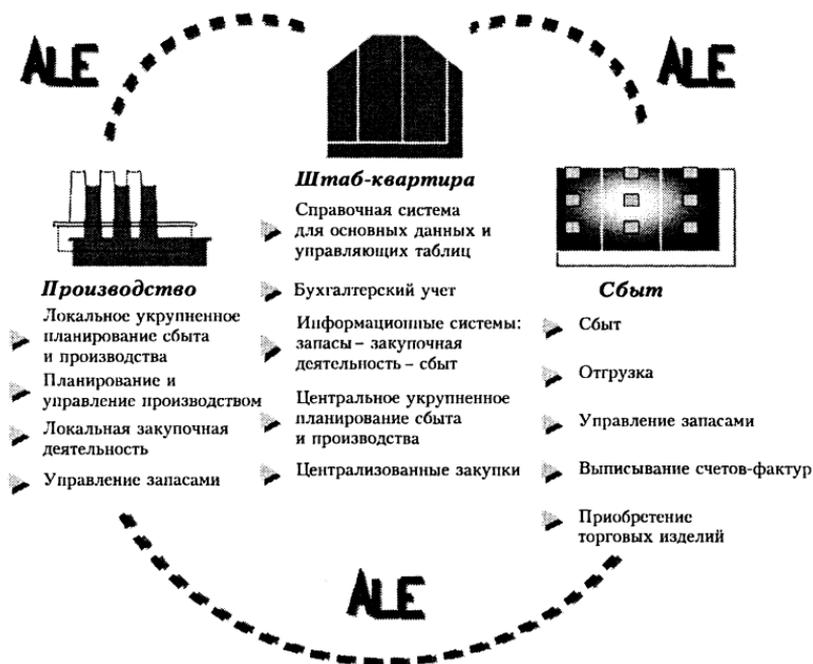


Рис.1.10. Интеграция между приложениями

необходимости учитывать другие функционирующие системы. ALE-технология обеспечивает управляемое взаимодействие между хозяйственными процессами в различных функционирующих системах. И последнее, но не менее важное: прикладные системы, разработанные третьей стороной (системы, разработанные не фирмой SAP), или системы R/3 можно также интегрировать через ALE без какой-либо фрагментации носителя данных.

ALE-концепция основывается на управляемом обмене сообщениями между хозяйственными процессами с непротиворечивым хранением данных. Приложения интегрируются посредством механизмов синхронной и асинхронной связи способом, управляемым во времени. Таким образом, ALE позволяет обеспечивать интеграцию хозяйственных процессов между различными системами. Одной из основных задач ALE-технологии является синхронизация хозяйственных процессов задействованных систем: здесь тоже согласовано используются стандартизированные процедуры. В ALE-технологии используются стандартизированные «промежуточные документы» для обмена сообщениями, которые включают в себя данные приложений (по тем же самым линиям связи функционирует EDI-интерфейс).

Посредством ALE фирма SAP делает доступной не только саму технологию, но также сценарии использования ALE-технологии вместе с системой R/3. Эти сценарии были разработаны с учетом того, что может потребоваться в хозяйственной деятельности. Ими можно воспользоваться для ускорения внедрения ALE-технологии.

Естественно, Базисная технология SAP также поддерживает обмен данными с приложениями других поставщиков на основе принятых стандартов: для того чтобы без помех передавать сообщения, используется электронный обмен данными (EDI), в основе которого лежит EDIFACT или ANSI X12. Эти сообщения могут состоять из заказов, счетов-фактур или других документов, касающихся хозяйственной деятельности. Таким образом, можно посылать и получать заказы или счета-фактуры электронным способом, это исключает необходимость вводить данные вручную и таким образом экономит время. Исключаются ошибки и не делаются испужными затраты.

Современная интеграция персональных компьютеров без какой-либо фрагментации носителей данных. Отдельные программы, предназначенные для работы на персональных компьютерах, можно эффективно использовать на мощных рабочих станциях и персональных компьютерах совместно с более крупными прикладными системами, как система R/3. Но это действительно повысит производительность: это тесная интеграция программ, предназначенных для работы на настольных компьютерах, с высоко интегрированным прикладным программным обеспечением.

Для обеспечения интеграции персональных компьютеров фирма SAP главным образом полагается на отраслевой стандарт OLE (Object Linking and Embedding). Кроме того, используются другие технологии, такие как RFC-технология, которые обеспечивают интеграцию широкого диапазона программных продуктов, предназначенных для работы на персональных компьютерах, имеющих на рынке.

Подходящие программные продукты, предназначенные для работы на настольных компьютерах, могут быть интегрированы в стандартное программное обеспечение, предназначенное для решения задач хозяйственной деятельности, там, где это требует заведенный порядок работы, и при этом достигается высокий уровень качества и нет необходимости фрагментации носителей информации. В этом случае для обработки данных, полученных из системы R/3 или из каких-либо других систем, пользователи могут использовать программные продукты, с которыми они привыкли работать, такие как текстовые процессоры или электронные таблицы. Это повышает их производительность и приводит к повышению качества их труда (рис.1.11).

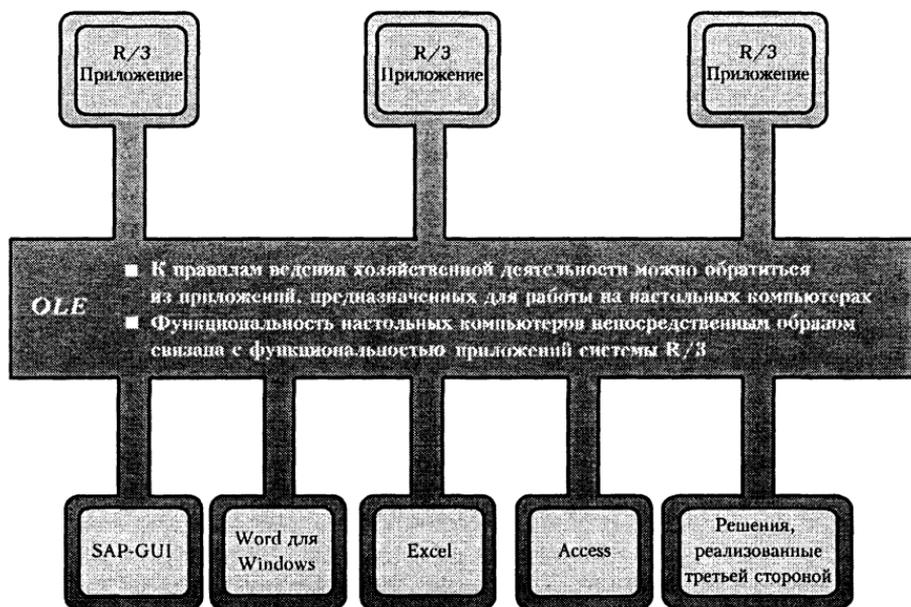


Рис.1.11. Пример OLE-технологии

Поток бизнес-операций SAP для интеграции всей установившейся практики хозяйственной деятельности в организации. С целью разумного объединения различных приложений и технологий для пользователя рекомендуется осуществить интеграцию более высокого уровня, ориентированную на установившуюся практику хозяйственной деятельности. Базисная технология SAP предоставляет для этой цели Поток бизнес-операций SAP (рис.1.12).

Поток бизнес-операций SAP связывает информацию с определенными хозяйственными процессами и делает ее доступной для ответственных работников в нужное время и в наилучшей возможной форме. Это приносит значительную прибыль.

Например, типичный поток бизнес-операций может включать в себя обработку заказа клиента от поступления заказа до доставки и выписывания счет-фактуры. Рассматриваемые события связаны друг с другом и являются последовательными компонентами хозяйственного процесса. Сотрудники, использующие приложения системы R/3, в любой момент могут выяснить статус данной транзакции и также могут обратиться к предыстории рабочего процесса.

Базисная технология SAP обеспечивает необходимый непрерывный поток информации и предоставляет интерфейсы, необходимые для обеспечения текущего контроля и описания этих процессов по

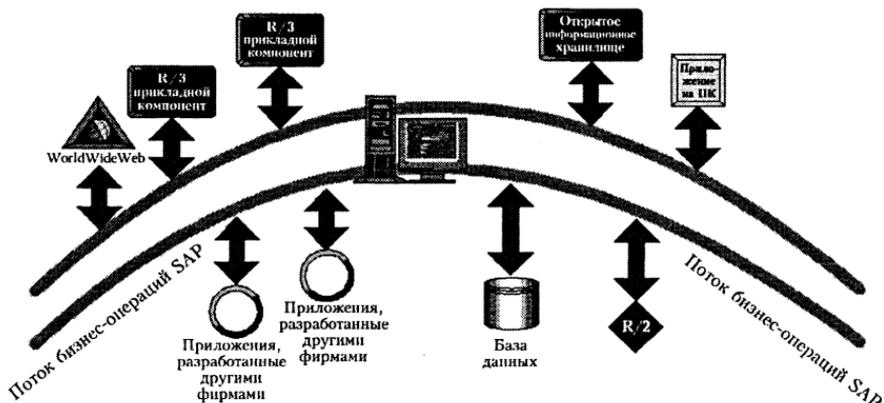


Рис.1.12. Поток бизнес-операций SAP

нескольким транзакциям. Существующие интерфейсы гарантируют беспрепятственную интеграцию документов (например, через оптические архивы) или информации из других приложений (например, EDI) для того, чтобы полностью смоделировать цепочку процессов.

Например, заказ, полученный через EDI, может запустить процесс потока операций. Данный заказ автоматически присваивается лицу, назначенному для этого процесса. Затем данное лицо вводит недостающую информацию и/или утверждает заказ. После этого данная транзакция – еще раз автоматически – передается следующему сотруднику в линейном подразделении фирмы: для подготовки документа о поставке, например. Здесь также данные добавляются в эту транзакцию. Сотрудники могут сконцентрировать внимание на своих текущих задачах, и им нет необходимости беспокоиться о передаче документов или о том, кто заменит заболевшего коллегу. Это обеспечивает эффективную обработку транзакций; время используется для выполнения фактической производительной работы. Конечно, поток бизнес-операций SAP также позволяет воспользоваться такими методами, как ALE и OLE, и объединить все связанные с этими методами возможности в процесс потока операций.

ABAP/4 Инструментальные средства разработки – это эффективный, интегрированный набор инструментальных средств разработки в среде клиент/сервер четвертого поколения. Эти средства поддерживают весь жизненный цикл разработки программных решений на базе архитектуры клиент/сервер, предназначенных для работы в рамках всей компании на основе быстрого макетирования. Графические интерфейсы пользователя и удобное манипулирование

объектами посредством объектно-ориентированной навигации делают данные инструментальные средства простыми в использовании.

Все объекты разработок, такие как модули на языке АВАР/4, ДИНПРО и модели данных хранятся в R/3-репозитории, который способствует быстрому и легкому доступу ко всем объектам. Он позволяет осуществлять подробный анализ использования данных во всей системе и обеспечивает высокую степень прозрачности в структурах данных.

С точки зрения разработчика, центральным информационным средством является активный АВАР/4-словарь данных. Он управляет всеми данными, связанными с приложениями, и обеспечивает, чтобы данная информация была постоянно доступна для интегрированных инструментальных средств разработки программ. Имеется также библиотека, содержащая ряд многократно используемых программных модулей, которые способствуют реализации рационального подхода к разработке программ в системе R/3 даже при удовлетворении конкретных требований клиентов.

Как доказывает практическое использование АВАР/4 Инструментальных средств разработки в системе R/3, они позволяют осуществлять разработку программного обеспечения группой разработчиков и при этом обеспечивается надежное управление версиями. Интегрированное средство Workbench Organizer (Организатор инструментальных средств) регистрирует и документирует все стадии в процессе разработки программ и обеспечивает, чтобы все изменения в программе правильно координировались.

Дополняют спектр услуг, предоставляемых АВАР/4 Инструментальными средствами разработки, инструментальные средства обеспечения качества и оптимизации работы системы, которые помогают провести анализ интерфейса конкретных прикладных программ с базой данных или всей системы.

Приложения, разработанные с помощью АВАР/4 Инструментальных средств разработки, могут быть перенесены для работы на многих различных платформах под управлением различных операционных систем с различными базами данных и графическими интерфейсами пользователя. Для программистов процесс связи с другими системами является прозрачным, и он надежно управляется инфраструктурой SAP-Базиса.

1.4.2. Пример разработки модели интегрированного управления производством

Методы, средства, объект исследований. Понятие Жизненного Цикла Изделия (ЖЦИ) являлось центральным при разработке модели интегрированного управления производством. Фазы ЖЦИ и

задачи, решаемые в каждой из фаз, были использованы для проведения декомпозиции задачи, являющейся главной для производств любого типа, в том числе и для машиностроительных предприятий, – выпуска продукции. В данном случае в качестве Изделия выступает любое машиностроительное изделие, так как оно обладает всеми признаками сложного технического объекта.

Апробация данного подхода и проверка правильности полученных результатов осуществлялась при обследовании организационно-функциональной структуры одного из предприятий. Данное предприятие занимается выпуском машиностроительной продукции промышленного назначения.

Для представления исследуемой информации был использован аппарат функциональных диаграмм IDEF, являющийся инструментом функционально-структурного анализа сложных систем. В соответствии с фазами ЖЦИ организационная структура предприятия была разделена на условные подразделения (см. рис.1.13), которые послужили основой для анализа соответствия их реальным подразделениям, построению модели интегрированного управления предприятием.

В действительности однозначного соответствия фаза \leftrightarrow подразделение не удастся достичь в силу более сложной структуры информации, необходимой для реализации всех функций предприятия.

Фаза «Использование и утилизация изделия» подразумевает участие нескольких субъектов (они могут быть как частными лицами, так и организацией), один из которых является конечным потребителем данного Изделия, а другой занимается утилизацией этого Изделия после окончания срока его службы. Оба субъекта, в частном случае, могут быть единым лицом. В любом случае они не входят в

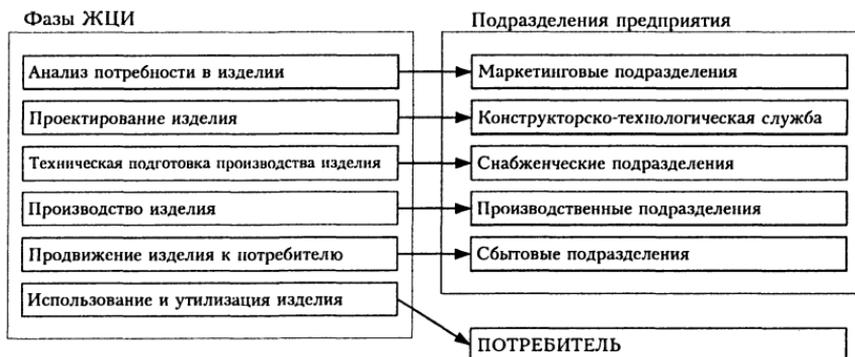


Рис.1.13. Соответствие фаз ЖЦИ организационной структуре

состав предприятия, производящего Изделие, поэтому подразделение «Потребитель» вынесено за пределы структуры предприятия и в исследовании будет участвовать как внешний объект.

Функциональный анализ. Декомпозиция задачи «Выпуск Продукции». В результате анализа организационной структуры предприятия была получена функционально-структурная диаграмма первого уровня, представляющая декомпозицию задачи «Выпуск Продукции» (см. рис.1.14). Рассматриваемая структура подразделений (Центров) отражает состав предприятия самого верхнего уровня иерархии организационных единиц. Путем проведения соответствия между диаграммой и фазами ЖЦИ были выявлены недостатки. Основной недостаток заключается в следующем.

Данное разбиение предприятия на Центры обусловлено минимизацией числа управляемых объектов в целях повышения качества управления, а возможность такой минимизации обусловлена небольшими размерами подразделений, подчиненных Центрам. Однако такой подход не даст представления о потоках объектов между фазами ЖЦИ, так как некоторые Центры покрывают функции нескольких фаз.

В целях проведения анализа взаимосвязей и потоков объектов между фазами ЖЦИ был составлен второй вариант декомпозиции задачи «Выпуск Продукции» (см. рис.1.15), которой позволил уточнить соответствие фаза \longleftrightarrow подразделение:

Анализ потребности в изделии \rightarrow Служба маркетинга

Проектирование изделия \rightarrow Научно-технический центр

Техническая подготовка производства изделия \rightarrow Службы технической подготовки изделия

Производство изделия \rightarrow Центр производства продукции

Продвижение изделия к потребителю \rightarrow Коммерческая служба

Функция «Службы маркетинга». Служба маркетинга (см. рис.1.16) решает задачи фазы ЖЦИ «Анализ потребности в изделии».

Входные объекты: статистика продаж, маркетинговая информация. Статистика продаж поступает из подразделения Коммерческая служба. Маркетинговая информация вырабатывается внутри Службы маркетинга.

Выходные объекты: Техническое задание на новое изделие, прогноз продаж.

Техническое задание на новое изделие предназначено для подразделения Научно-технический центр, содержит информацию, необходимую для разработки новых изделий, формируется на основе маркетингового анализа потребностей рынка. Прогноз продаж содержит информацию о количественном и номенклатурном составе изделий, рекомендуемых для производства на свободную продажу, формируется на основе прогноза продаж и маркетинговой информации.

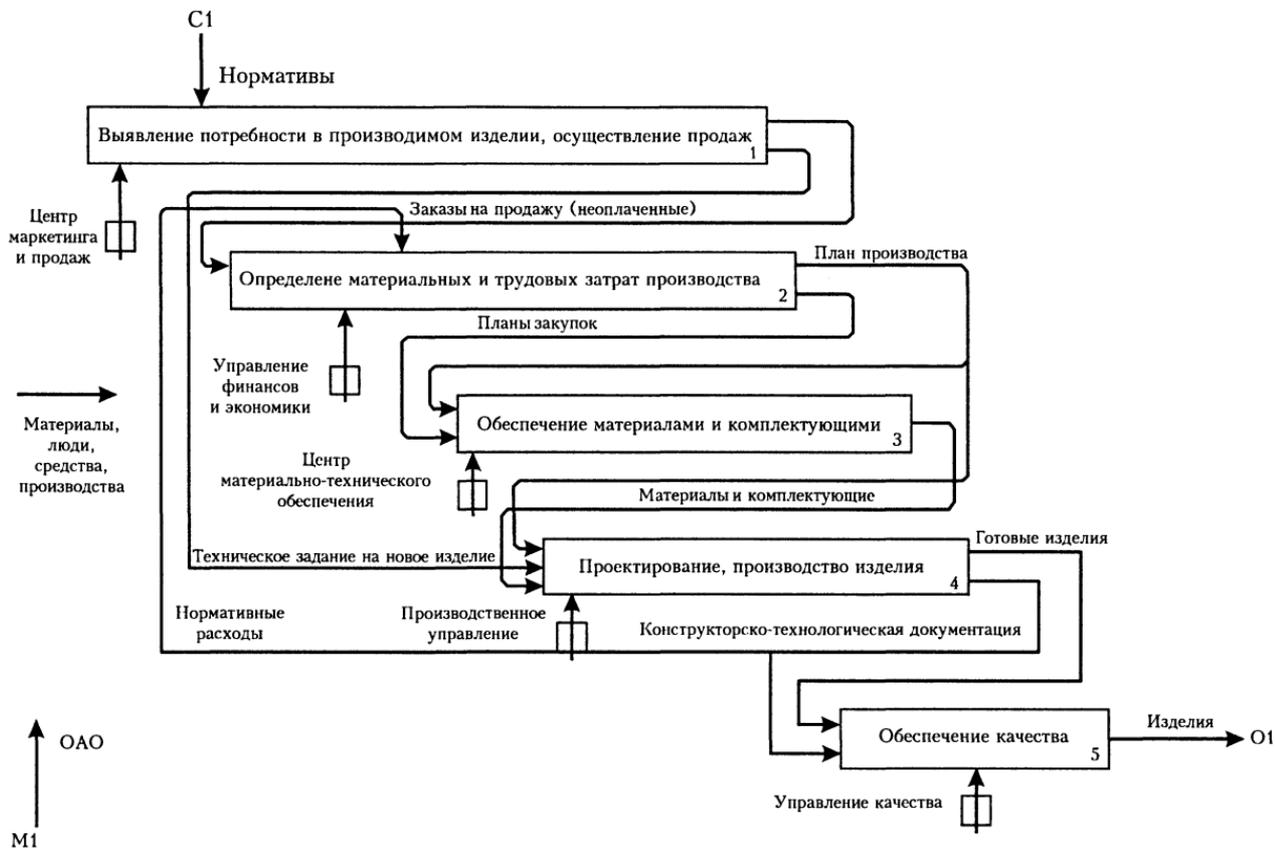


Рис.1.14. Первый вариант декомпозиции задачи «Выпуск Продукции»

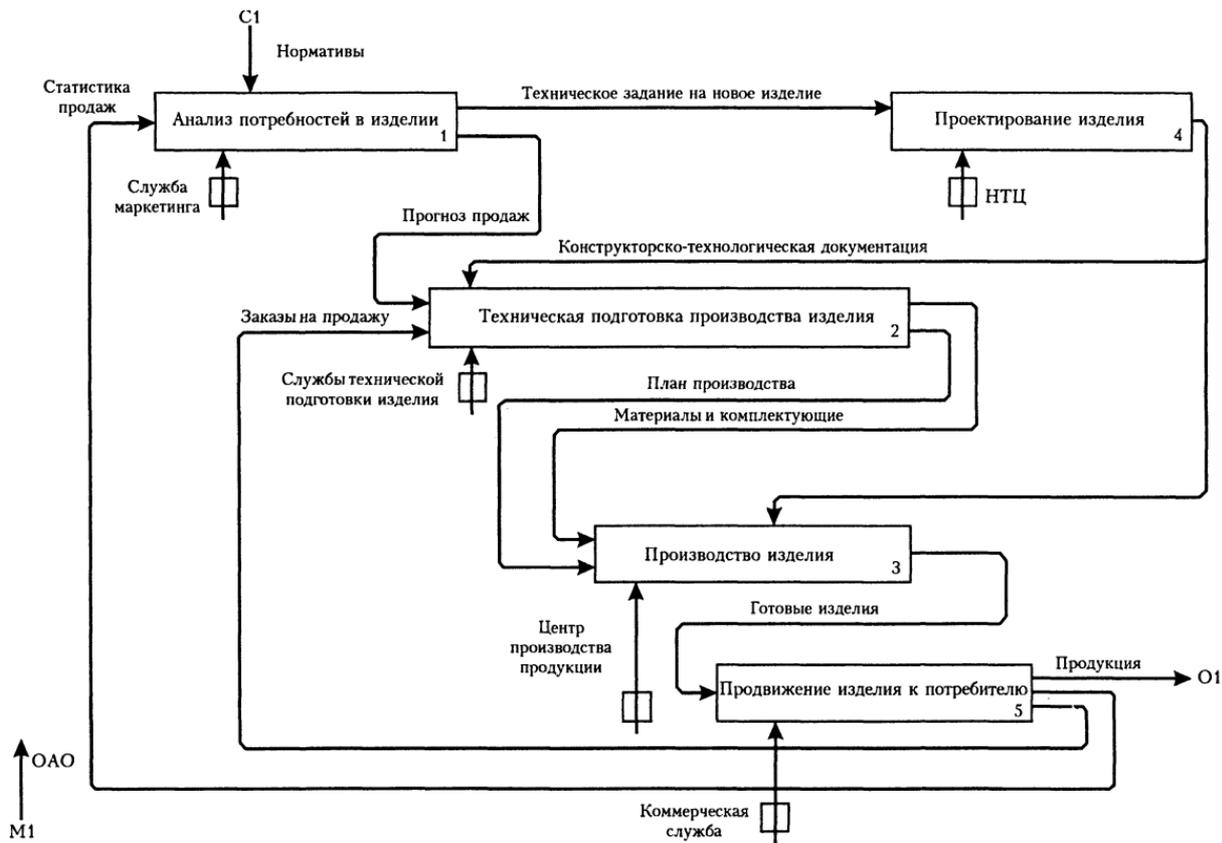


Рис.1.15. Второй вариант декомпозиции задачи «Выпуск Продукции»

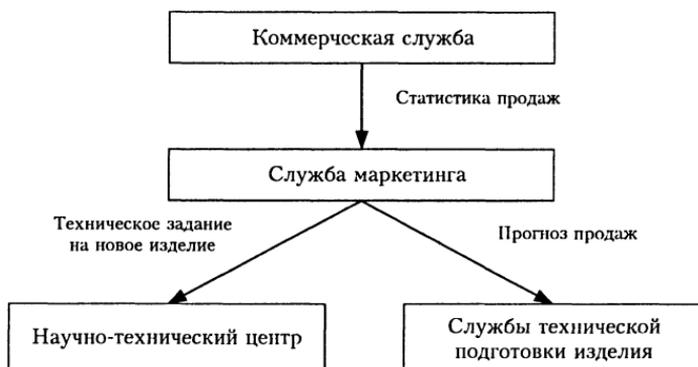


Рис.1.16. Структура организационной единицы «Службы маркетинга»

Функция «Научно-технического центра». Научно-технический центр (см. рис.1.17.) решает задачи фазы ЖЦИ «Проектирование изделия».

Входные объекты: Техническое задание на новое изделие.

Техническое задание поступает из Центра маркетинга.

Выходные объекты: Конструкторско-технологическая документация.

Конструкторско-технологическая документация содержит информацию о конструкции изделия и технологии его изготовления, разрабатывается на основе Технического задания и передается в Службы технической подготовки изделия и Центр производства продукции.

Функция Служб технической подготовки. Службы технической подготовки являются не организационной единицей, а

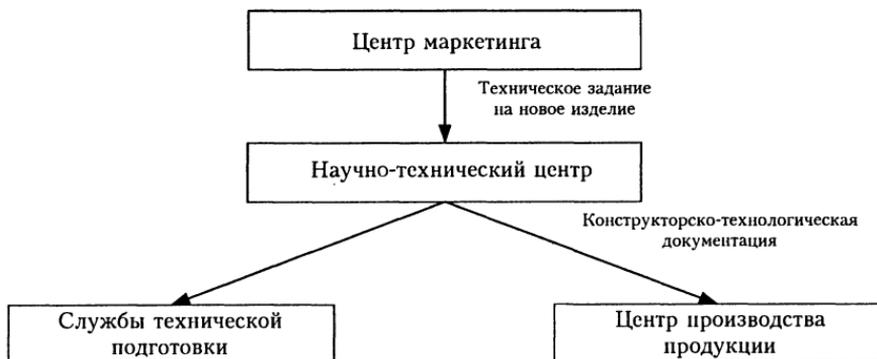


Рис.1.17. Структура организационной единицы «Научно-технический центр»

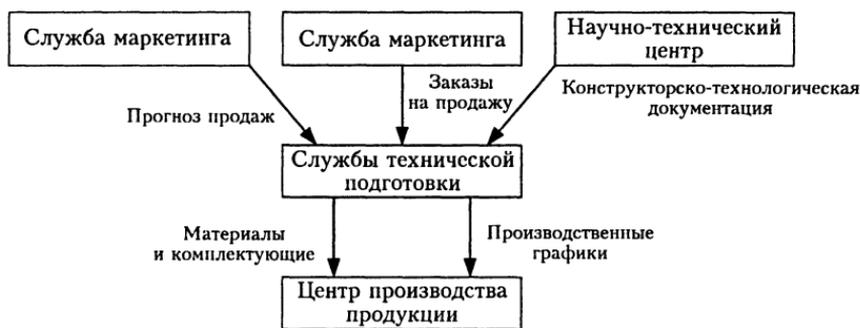


Рис.1.18. Структура Служб технической подготовки

логическим обобщением подразделений, которые решают задачи фазы ЖЦИ «Техническая подготовка производства изделия» (см. рис.1.18), и состоят из следующих подразделений различных Центров: Службы экономики, Службы закупок, Службы начальника производства.

Входные объекты: прогноз продаж, заказы на продажу, конструкторско-технологическая документация.

Прогноз продаж поступает из Службы маркетинга, заказы на продажу из Коммерческой службы, конструкторско-технологическая документация из Научно-технического центра.

Выходные объекты: план производства, материалы и комплектующие.

План производства содержит информацию о количественном и номенклатурном составе изделий, принятых к производству на текущий период, составляется на основе прогноза продаж и заказов на закупку Службой экономики. На основе плана производства и нормативной информации из Конструкторско-технологической документации формируются планы закупок материалов и комплектующих, предназначенные для Службы закупок, которая осуществляет их доставку от внешних поставщиков. Выходные объекты предназначены для Центра производства продукции.

Функция «Центра производства продукции». Центр производства продукции (см. рис.1.19) решает задачи фазы «Производство изделия».

Входные объекты: план производства, материалы и комплектующие.

План производства и материалы и комплектующие поступают в Службу технической подготовки.

Выходные объекты: готовые изделия.



Рис.1.19. Структура организационной единицы «Центр производства продукции»

На основе плана производства разрабатываются производственные графики. На основе управляющей информации, содержащейся в производственных графиках, из материалов и комплектующих изготавливаются конечные изделия и передаются в ведение Коммерческих служб.

Функция «Коммерческой службы». Коммерческая служба (см. рис.1.20) решает задачи фазы ЖЦИ «Продвижение изделия к потребителю».

Входные объекты: готовые изделия.

Готовые изделия поступают из Центра производства продукции.

Выходные объекты: продукция.

Отгрузка и продажа продукции для конечного потребителя.

Интегрированная система управления подразумевает наличие единой программно-информационной среды, поддерживающей все

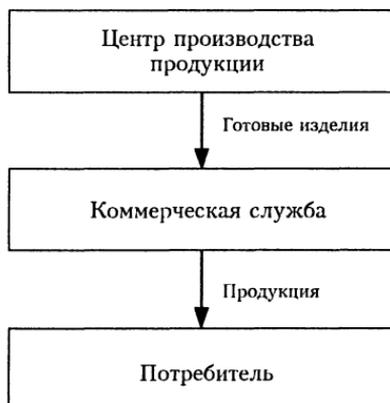


Рис.1.20. Структура организационной единицы «Коммерческая служба»

аспекты функциональной деятельности предприятия, и должна поддерживать автоматизированное решение всех основных задач, входящих в процесс управления жизненным циклом изделия.

Основные задачи каждой из фаз ЖЦИ были рассмотрены выше. Проведем анализ и классификацию функциональной деятельности предприятия с точки зрения функций управления жизненным циклом изделия.

Декомпозиция фаз по функциям управления ЖЦИ. В число функций управления ЖЦИ входят:

- менеджмент;
- обеспечение качества;
- разработка;
- изготовление;
- сопровождение;
- документирование;
- обслуживание.

Каждая функция имеет свою интерпретацию для каждой фазы ЖЦИ.

Рассмотрим интерпретацию этих функций для фазы Технической подготовки производства.

Службы технической подготовки производства имеет следующие основные виды деятельности:

- 1) проектирование;
- 2) анализ проекта;
- 3) инженерное тестирование и моделирование;
- 4) анализ вносимых изменений;
- 5) администрирование проекта;
- 6) технологическое проектирование;
- 7) планирование ресурсов.

К функции менеджмента можно отнести деятельность 5; обеспечение качества – 2; разработка – 1, 6; изготовление – 3; сопровождение – 4; документирование присутствует во всех видах деятельности, обслуживание – 7.

Декомпозиция функций по уровням. В свою очередь каждый вид деятельности может быть разделен на несколько уровней. Например, проектирование:

- концептуальное проектирование;
- инженерное проектирование;
- детальное проектирование.

Управление предприятием имеет следующие уровни:

- планирование;
- диспетчирование;

- оперативное управление;
- управление оборудованием.

Система планирования имеет следующую иерархическую структуру:

- перспективное планирование;
- текущее планирование;
- оперативное планирование.

Требования к интегрированной системе управления. Приведенная выше классификация функциональной деятельности предприятия затрагивает все производственные ресурсы. Это означает, что все объекты – как материальные (производственные мощности, материалы, инструмент, люди, источники энергии, транспорт и т.д.), так и информационные (модели конструкций, описания технологических процессов и процессов управления, расписания, производственные задания, отчеты и т.д.) – управляются через единую информационную среду, описываются в ней едиными средствами, доступны на любом уровне иерархии.

Таким образом, можно сформулировать требования к интегрированной системе управления:

- обеспечение информационно-организационной инфраструктуры системы управления;
- обеспечение среды существования всех видов информационных объектов, поддерживающей их целостность и возможность взаимодействия;
- обеспечение средств описания объектов, доступных на любом уровне иерархии;
- предоставление единых средств доступа ко всем объектам, независимо от их физического расположения.

Резюме. Требование единой информационной и организационной инфраструктуры является необходимым, так как интегрированная система управления представляет собой симбиоз автоматизированного производства и автоматизированного учреждения, потоки информации и управления в ней общие, а значит, необходима и общая инфраструктура для обмена информацией и управлением между компонентами системы.

1.5. Взаимосвязь технологий и стандартов в области CALS

Принятие стратегических документов в области обеспечения качества и повышения конкурентоспособности товаров и услуг (например, глобальная концепция обеспечения качества в ЕС) и связанная с этим необходимость для поставщика товаров и услуг доказывать свою способность предоставлять качественные товары и услуги

привело к стандартизации основных информационных технологий управления предприятием, разработки продукции, информационного обмена в цепочке «субпоставщик – производитель – потребитель». Место и роль таких информационных технологий и международных стандартов, а также взаимосвязь между ними, приведены на рис. 1.21. Суть этих технологий кратко изложена ниже.

Выходы, связанные с производством продукции как у поставщика, так и у производителя можно представить при использовании стандартов MRP, MRP II, ERP, ISO 15531 ManDate.

Характеристики продукции и ее состояния как у поставщика, так и у производителя можно представить при использовании стандартов ISO10303 STEP, ISO 15531 ManDate.

Требования потребителя и производителя учитываются при использовании ФСА, ФФА, FMEA, QFD.

Обратная связь между потребителем и производителем, а также между производителем и субпоставщиком может быть организована на базе стандартов ISO 9000, MRP, MRP II, ERP, ISO 15531 ManDate, ISO 10303 STEP.



Рис.1.21. Роль и место международных стандартов для повышения эффективности управления в цепочке «поставщик – производитель – потребитель»

ISO 9000 – семейство стандартов на системы качества предприятия. Система качества – часть системы управления предприятия, охватывающая основные бизнес процессы (в настоящее время более 20 процессов). Разработана в середине 1980-х годов как обобщение передового опыта по обеспечению качества и воплощение Глобальной Европейской концепции в области качества. Предназначена для решения следующих основных задач:

- обеспечения климата доверия в экономике;
- предоставления потребителю объективных доказательств способности поставщика к производству товаров и услуг определенного уровня качества;
- повышения конкурентоспособности предприятий.

Система качества является наиболее распространенным стандартом за всю историю ISO, их используют несколько сот тысяч предприятий практически во всех странах мира. Соблюдение требований стандарта в настоящее время рассматривается как пропуск на международный рынок товаров и услуг. В России с 1998 г. соблюдение требований ISO 9000 – обязательное условие для получения госзаказа (постановление Правительства РФ №113 от 02.02.1998 г.).

MRP – стандарт на планирование материальных ресурсов (Material Requirements Planing), первый из серии стандартов на планирование материальных ресурсов, разработан в 1960-х годах, обеспечивает согласование действий снабженческих, производственных и сбытовых подразделений по формированию заказов в реальном масштабе времени и материального учета. Не поддерживает нулевых производственных запасов и потому не обеспечивает поставок в режиме just in time (точно в срок).

Одним из наиболее распространенных методов управления производством в мире является стандарт MRP II (Manufacturing Resource Planning), разработанный в США и поддерживаемый американским обществом по контролю за производством и запасами – American Production and Inventory Control Society (APICS). MRP II – это набор проверенных на практике разумных принципов, моделей и процедур управления и контроля, служащих повышению показателей экономической деятельности предприятия.

С середины 1990-х годов стандарт MRP II применяется для планирования потребностей в распределении и ресурсах на уровне предприятия – Enterprise Resource Planning, а интегрированные программные продукты, обеспечивающие такое планирование, называются ERP-системами (например, SAP R3, BAAN, MGF/PRO, Oracle Application).

Как известно, система класса MRP II имеет целью электронное моделирование всех основных процессов, реализуемых

предприятием, таких как снабжение, запасы, производство, продажа и дистрибуция, планирование, контроль за выполнением плана, затраты, финансы, основные средства и т.д. Следует отметить, что Международный стандарт по управлению качеством процессов ISO 9000 обязывает иметь на предприятии указанные модели, хотя и не требует их электронной реализации.

ERP – дальнейшее развитие стандарта на организацию производства и материально-технического снабжения (Enterprise Resource Planing) – разработан в 1990-х годах. Поддерживает концепцию CIM (компьютеризованного интегрированного производства) и оптимального управления логистическими потоками в реальном масштабе времени, поставки в режиме just in time (точно в срок).

В настоящее время развивается в концепции DRP (Dynamical Resource Planing) – организации производства динамической конфигурации, в которой бизнес процессы могут оптимально изменяться, в зависимости от изменения задач. Поддерживает концепции глобализации бизнеса, работы в режиме 24×365 и т.д.

В последние десятилетия одним из главных факторов экономического роста промышленно развитых стран стало массовое внедрение информационных технологий в производственные процессы. Такие технологии призваны обеспечить сокращение финансовых и временных затрат при производстве сложной наукоемкой продукции. Для информационной интеграции процессов, протекающих в ходе жизненного цикла продукции, была разработана CALS-идеология, реализованная позже в виде CALS-технологий.

Ситуация на мировом рынке наукоемкой продукции развивается в сторону полного перехода на безбумажную электронную версию проектирования, изготовления и сбыта продукции. По прогнозам зарубежных специалистов, в ближайшее время на внешнем рынке невозможно будет продать машинотехническую продукцию без соответствующей международным стандартам безбумажной электронной документации. Передовые зарубежные фирмы рассматривают работу в этом направлении как действенное средство ограничения доступа на международный рынок наукоемкой продукции тех стран, которые не сумеют своевременно освоить соответствующие международным требованиям безбумажные электронные технологии. Предположительно в ближайшие несколько лет мировой рынок наукоемких технологий – так же как и рынок промышленной кооперации – полностью перейдет на стандарты CALS.

Аббревиатура CALS в ходе своего развития имела целый ряд интерпретаций:

- Computer-Aided Acquisition and Logistics Support – компьютеризированная поддержка логистических систем – 1980 г.;

- Computer-Aided Acquisition and Support – компьютеризированные поставки и поддержка – 1988 г.;
- Computer-Aided Acquisition and Lifecycle Support – поддержка непрерывных поставок и жизненного цикла – 1993 г.;
- Commerce At Light Speed – Бизнес в высоком темпе – 1998 г.

Что понимают под CALS в настоящее время:

- CALS – это идеология создания единой информационной среды для процессов проектирования, производства, испытаний, поставки и эксплуатации продукции в ходе всех стадий жизненного цикла – от замысла до утилизации;
- интеграция достигается путем стандартизации представления информации (или, скажем, результатов) в процессах проектирования, материально-технического снабжения, производства, ремонта, послепродажного сервиса и т.д. Такой подход создает новый базис для информационной интеграции и преемственности в использовании информации;
- эффективный бизнес сегодняшнего дня имеет явную тенденцию к географической распределенности. Компании кооперируются для того, чтобы вместе выполнить сложный проект или вывести на рынок новый продукт. Возникают так называемые виртуальные предприятия – форма объединения на контрактной основе предприятий и организаций, участвующих в поддержке жизненного цикла общего продукта и связанных общими техническими процессами. Этот сложный организм должен жить по единым правилам в едином информационном пространстве, позволяющем непосредственно использовать обмен данными в электронной форме между партнерами. В случае изменения состава участников – смены поставщиков или исполнителей – обеспечиваются преемственность и сохранность уже полученного продукта (моделей, расчетов, документации, программ, баз данных).

Таким образом CALS-технологии представляют собой современную организацию процессов разработки, производства, послепродажного сервиса, эксплуатации изделий путем информационной поддержки процессов их жизненного цикла на основе стандартизации методов представления данных на каждой стадии жизненного цикла и безбумажного электронного обмена данными. Концепция CALS определяет набор правил, регламентов, стандартов, в соответствии с которыми строится информационное («компьютерное») взаимодействие участников процессов проектирования, производства, испытаний и т.д.

Целью применения CALS-технологий как инструмента организации и информационной поддержки всех участников создания, производства и пользования продуктом является повышение эффективности их деятельности за счет ускорения процессов исследования и разработки продукции, придания изделию новых свойств, сокращения издержек в процессах производства и эксплуатации продукции, повышения уровня сервиса и технического обслуживания. В современных условиях CALS-технологии служат важнейшим инструментом повышения эффективности бизнеса, конкурентоспособности и привлекательности продукции.

По данным западных аналитиков, применение CALS-технологий позволяет, например, в масштабах промышленности США экономить десятки миллиардов долларов в год, сократить сроки проведения всех работ на 15–20%.

CALS-технологии активно применяются прежде всего при разработке и производстве сложной наукоемкой продукции, создаваемой интегрированными промышленными структурами, включающими в себя НИИ, КБ, основных подрядчиков, субподрядчиков, поставщиков готовой продукции, потребителей, предприятия технического обслуживания, ремонта и утилизации продукции. Вместе с тем применение CALS-технологий позволяет эффективно, в едином ключе решать проблемы обеспечения качества выпускаемой продукции, поскольку электронное описание процессов разработки, производства, монтажа и т.д. полностью соответствует требованиям международных стандартов ISO серии 9000, реализация которых гарантирует выпуск высококачественной продукции.

Ни одна из известных ERP-систем не включает в свой состав ряд необходимых для CALS компонентов, таких как САПР, генератор программ с ЧПУ, инженерные расчеты, организация производства на уровне цеха.

ISO 15531 ManDate – стандарты из системы стандартов CALS – технологий. Предназначен для обеспечения коллективного доступа поставщика и потребителя к информации о производственном процессе поставщика. Использует согласованные со стандартом ISO 10303 STEP форматы представления данных.

ISO 10303 STEP – основное семейство стандартов из системы стандартов CALS-технологий (в настоящее время включает около сотни стандартов и проектов). Предназначен для обеспечения коллективного доступа поставщика и потребителя к информации о:

- конструкции изделия;
- процедурам испытаний изделия;
- эксплуатационной документации на изделие;
- другой информации по всем стадиям жизненного цикла изделия.

Разработан в конце 1980-х годов МО США при участии Министерства торговли США и предназначался первоначально для обеспечения поставок военной техники и технологий. В настоящее время все шире охватывает невоенные области, прежде всего машиностроение и промышленное строительство.

Важность управления данными об изделии, представленными в формате ISO 10303 STEP, связано со следующими обстоятельствами. Данные о конструкции изделия занимают значительную часть в общем объеме информации, используемой в ходе его жизненного цикла (ЖЦ). На основе этих данных решается ряд задач производства изделия, материально-технического снабжения, сбыта, эксплуатации, ремонта и др. (рис.1.22). Сегодня, несмотря на широкое

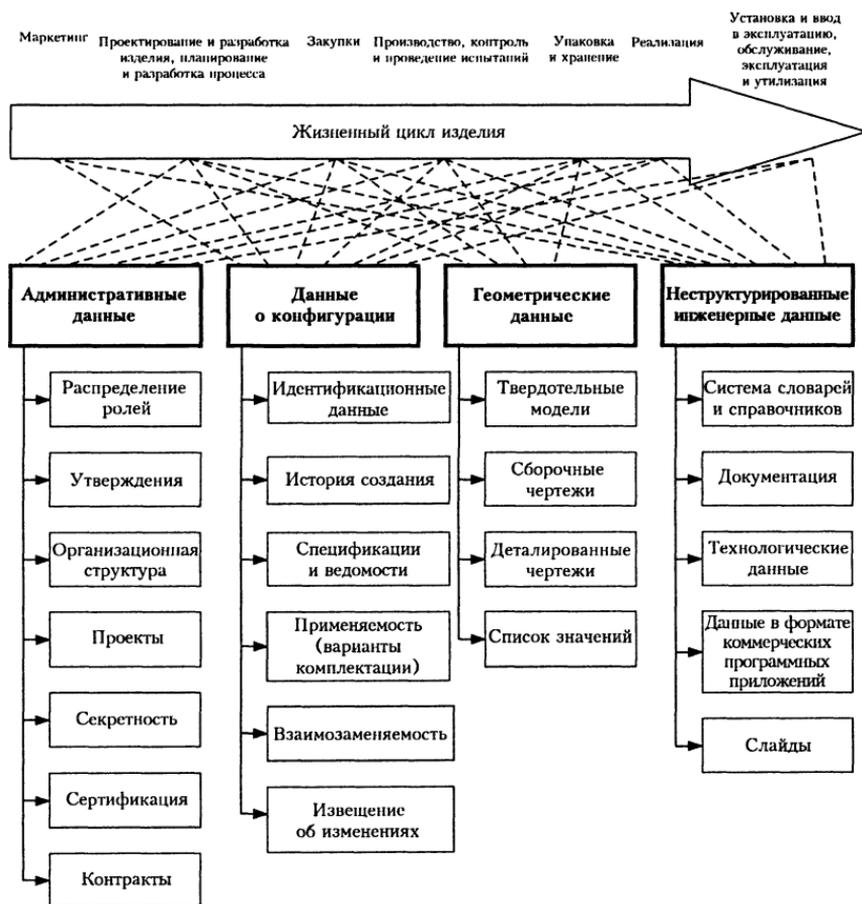


Рис.1.22. Использование конструкторских данных в ходе ЖЦ изделия

использование компьютерных технологий, процессы проектирования и производства промышленной продукции в основном выполняются традиционными методами на основе бумажных документов. Объем проектных работ, выполняемых с использованием систем автоматизированного проектирования, незначителен, причем, как правило, полученные результаты все равно переводятся из цифрового вида в форму бумажных документов.

Одна из причин такого перевода – сложность интеграции результатов проектирования, полученных разными методами. В результате, информация о сложном машинотехническом изделии, таком как автомобиль или самолет, представляет собой тонны бумажной документации. При этом преимущества применения электронного представления информации об изделии, состоящие в возможности многократного использования информации, сокращении затрат на ее повторный ввод, подготовку и обработку, не используются в полной мере.

Электронная информационная CALS-интеграция производственных модулей, обеспечивающая безбумажную технологию на основе электронного документооборота (так называемых PDM-систем), базируется на таком важном понятии, как модель технологической среды. Модель технологической среды машиностроительного предприятия включает в себя:

- модели для инженерных расчетов, конструирования и автоматизированного проектирования изделий;
- модели построения обрабатываемых поверхностей и генерация программ ЧПУ;
- модели технологической последовательности обработки деталей;
- модели для оптимального планирования производства.

ФСА – функционально-стоимостной анализ – технология разработки и анализа продуктов, позволяющая сократить себестоимость продуктов на основе выравнивания соотношения «важность – стоимость» элементов продукции. Разработан в США в конце 1940-х годов, принят как стандарт большинством развитых стран в конце 1960-х.

ФФА – функционально-физический анализ – технология разработки и анализа технических систем, позволяющая разрабатывать продукты, реализующие эффективные принципы действия. Разработан в СССР в конце 1970-х – начале 1980-х годов, в настоящее время достаточно широко внедряется в развитых странах бывшими советскими специалистами.

FMEA – анализ (Failure mode and effect analysis) – анализ причин и последствий дефектов для потребителей – метод анализа продуктов и процессов, позволяющий выявить элементы конструкции (анализ продуктов) или операции процессов (анализ процессов), имеющие повышенный потенциальный риск для потребителя и

разработать предупреждающие мероприятия, снижающие риск до приемлемых величин. Разработан рядом авиакосмических фирм США в рамках программы полета к Луне НАСА в середине 1960-х годов. В настоящее время является фактическим стандартом в большинстве развитых стран.

QFD (quality function deployment) – развертывание функций качества – технология разработки и подготовки производства продуктов, позволяющая эффективно преобразовывать запросы потребителя в технические требования. Использует ряд последовательно перестраиваемых таблиц – «домиков качества» – для всех стадий разработки и подготовки производства изделий. Разработана в 1970-х годах в Японии. В настоящее время широко применяется в большинстве развитых стран, где рассматривается как эффективное оружие в конкурентной борьбе.

Сейчас стало очевидным, что первый разработчик ERP-систем, который выйдет на отечественный рынок с предложениями интегрированных решений, обеспечивающих CALS-технологии и, следовательно, удовлетворяющих стандартам ISO 9000 и MRP II, получит безусловное преимущество.

1.6. Обучение CALS-технологиям

Авторы глубоко убеждены в том, что для систематического продвижения и сопровождения CALS-технологий в Российской Федерации, а также для интеграции в мировое CALS-сообщество необходима учебная специальность по CALS-технологиям. Эта специальность должна готовить системных CALS-аналитиков, а также специалистов в области интегрированных систем управления предприятиями.

Так как составление рабочего плана по данной специальности не входит в тематику данной книги, приведем в качестве примера программу курса «Системная интеграция», который является завершающим в цикле специальных дисциплин.

Рабочая программа курса «Системная интеграция»

Аннотация

Технология информационных систем все в большей и большей степени становится компонентом бизнес-стратегии предприятия. Это основа сбыта, управления финансами, обслуживания заказчика, всего производства и проектно-конструкторских работ. Сегодня информационная система должна связывать все функции управления предприятием и выполнения заказов, и функционировать на различных уровнях. Стратегия управления предприятием должна позволять сочетание нескольких производственных методов: производство на заказ, сборка на заказ или производство на склад в единой системе управления.

Темы лекционных занятий (40 лекций).

ТЕМА 1. Планирование

- 1.1. Бизнес-план/производственный план Бизнес-планирование позволяет переводить стратегические задачи бизнеса на язык конкретных производственных планов. Бизнес-план включает в себя производственные планы, планы управления запасами и планы потребностей рынка. На основе бизнес-плана разрабатывается объемно-календарный план, обеспечивающий баланс спроса и предложения.
- 1.2. Прогнозирование. Расчет прогноза спроса на основе данных предыстории спроса и оперативной информации с использованием статистических методов: полиномиальная регрессия; экспоненциальное сглаживание с анализом тренда; сезонные колебания с анализом цикличности; анализ временных рядов.
- 1.3. Укрупненное планирование ресурсов. Планирование критических производственных мощностей и критических материалов в условиях заказов с большим сроком исполнения.
- 1.4. Планирование потребностей в материалах на основе: прогноза реализации; независимых потребностей; производственных заказов; заказов на продажу; проектов для заказчиков.
- 1.5. Финансовый анализ. Анализ планируемого потока денежных средств на основе данных текущего плана потребностей в мощностях.

ТЕМА 2. Производство

Управление всеми фазами производственного цикла, интегрированное с функциями планирования в целях образования замкнутого цикла планирующих функций.

- 2.1. Производственное планирование. Преобразование объемно-календарного плана и плана потребностей в материалах в подробный план действий: календарное планирование, производственные задания, недельные и дневные графики, диаграммы Ганта.
- 2.2. Оперативное управление производством. Методы выдачи материалов в производство; отслеживание дефицита материалов; контроль работ по отдельным операциям или по завершению заказа; перепланирование; бракировка и исправление брака; субподрядные работы; отслеживание производственных затрат.
- 2.3. Учет рабочего времени. Методы регистрации затрат времени; графики работы; виды затрат рабочей силы; нормативные, плановые и фактические затраты времени.

- 2.4. Контроль себестоимости продукции. Наборы правил для управления нормами заработной платы, затрат на оборудование и накладных расходов; моделирование затрат для определения воздействия различных составляющих себестоимости; моделирование отпускной и розничной цены.

ТЕМА 3. Проектно-конструкторские работы

- 3.1. Управление чертежами. Категории чертежей и конструкторских спецификаций; применимость изделий; согласование конструкторской и технологической спецификаций; контроль по срокам и непротиворечивости данных.
- 3.2. Классификация продукции. Классификация изделий на основе характерных признаков.

ТЕМА 4. Управление запасами

Управление данными по запасам, интегрированное с функциями планирования.

- 4.1. Управление изделиями. Консолидированные данные по себестоимости; данные материально-технического снабжения; данные реализации; данные управления запасами; планировочные данные.
- 4.2. Управление складскими запасами. Типы складских операций; наличные запасы и периодическая инвентаризация; анализ движения запасов по методу ABC-классификации.
- 4.3. Статистическое управление запасами. Статистическое прогнозирование по методам: самый поздний, среднее использование, предыдущий год, экспоненциальное сглаживание; вывод правил составления заказов.
- 4.4. Управление размещением запасов. Организация хранения запасов и операции по их выдаче; правила выбора местонахождений для принимаемых материалов и заказов на выдачу; контроль условий хранения запасов с помощью правил и условий хранения; принципы LIFO и FIFO для операций выдачи; управление выдачей материалов по дате срока годности партии.
- 4.5. Планирование распределения. Методы ведения оптимального уровня запасов по распределенным складам; центры распределения и центры поставок; прогнозируемый и целевой уровень запасов.
- 4.6. Управление партиями. Контроль движения по партиям или серийным номерам; контроль прохождения материала через все производственные операции до окончательного использования.

ТЕМА 5. Управление реализацией

- 5.1. Предложения по реализации. Создание предложений; включение предложений в систему составления прогнозов и планирования; детальный анализ возможных прибылей и убытков; использование цен.
- 5.2. Контракты на реализацию. Ценообразование по временным периодам, по объемам, по изделиям и группам изделий; графики отгрузки;
- 5.3. Управление реализацией. Цены и скидки; контроль отклонений цен; обработка заказов на продажу; обработка позиций заказа; выборка и отгрузка; выписка счетов.
- 5.4. Маркетинговая информация. Затраты на продажу; анализ перспектив и результатов рекламы; статистика продаж; финансовые проводки.

ТЕМА 6. Управление снабжением

- 6.1. Предложения по снабжению. Ведение предложений; ценовая политика поставщиков; история предложений.
- 6.2. Контракты на снабжение. Фиксированные контракты и рамочные заказы; учет политики цен и скидок поставщиков при повторных заказах; револьверные поставки и поддержка режима «точно в срок»; история контрактов.
- 6.3. Управление снабжением. Графики снабжения; снабжение по контрактам, по проектам и электронные заказы на снабжение; типы заказов и процедуры обработки; поступления полные, частичные, допоставка; входной контроль; отслеживание поставщиков и хода снабжения.
- 6.4. Статистика по снабжению. История отгрузок и цен по изделиям и поставщикам.
- 6.5. Электронный обмен данными. Электронная торговля.

ТЕМА 7. Управление проектами

Управление производством, ориентированным на потребителя.

Выпуск продукции с исполнениями. Управление научно-исследовательскими и опытно-конструкторскими работами.

- 7.1. Финансовая подготовка проекта. Оценка стоимости; материальные затраты, операционные затраты, надбавки, общие расходы.
- 7.2. Управление проектами. Создание проекта и подпроектов; разработка проекта, изделия по проекту, проектные спецификации, маршрутизация по проекту.
- 7.3. Сетевое/модульное планирование по проекту. Планирование деятельности по проекту с использованием сетевых методов; график проекта и план загрузки мощностей.
- 7.4. Конфигуратор продукции.

ТЕМА 8. Финансы

- 8.1. Главная книга. Бухгалтерские операции по счетам для множества центров затрат и/или проектов; учет организаций типа холдингов с проведением операций между дочерними фирмами и консолидацией общей отчетности; субсчета; налогообложение; бюджет.
- 8.2. Управление центрами затрат и субучет. Категории затрат; распределение центров затрат.
- 8.3. Расчеты с заказчиками. Расширенное управление кредитами; финансовые группы; обработка счетов на реализацию; обработка платежей; прогноз ликвидности; списание со счета.
- 8.4. Расчеты с поставщиками. Обработка счетов поставщиков; финансовые группы; обработка платежей; списание со счета.
- 8.5. Финансовая отчетность и консолидация баланса
- 8.6. Учет основных фондов. Списание стоимости; три типа амортизации для фондовых книг, налогов и моделирования; периодическая книжная и налоговая амортизация; переоценка и амортизация по различной налоговой базе.

ТЕМА 9. Техобслуживание

- 9.1. Организация техобслуживания и ремонта
- 9.2. Обработка заказа на техпическое обслуживание
- 9.3. Управление возвращенной продукцией

Темы практических занятий**ТЕМА 1. Планирование**

- Составление бизнес-плана для семейства изделий.
- Разработка производственных планов для конечных изделий.
- Расчет производственных заказов.
- Расчет потребностей в закупках.

Тема 2. Управление производством

- Составление графиков серийного производства.
- Перепланирование производственных заказов.
- Проведение отпуска материалов в производство.
- Завершение выполнения производственных заказов.
- Расчет фактического рабочего времени.
- Расчет фактических затрат.

ТЕМА 3. Управление снабжением

- Расчет потребностей в материалах.
- Составление заказов на закупку.
- Составление поступлений по заказам на закупку.
- Оформление присмки.

ТЕМА 4. Управление реализацией

- Составление предложения на реализацию.
- Составление заказа на продажу.
- Расчет цен и скидок на продажу.
- Оформление отгрузки.

ТЕМА 5. Управление проектами

- Разработка структуры проекта.
- Разработка задач по проекту.
- Составление сетевого плана проекта.
- Планирование работ по проекту.
- Разработка бюджета проекта.
- Выполнение и закрытие проекта.

Литература

Горшков А.Ф., Соломенцев Ю.М. Применимость реберных замещений в классе комбинаторных задач на графах // Доклады АН. 1994. Т.337, №2. С.151–153.

Дмитров В.И. Опыт внедрения CALS за рубежом // Автоматизация проектирования. 1997. №1. С.3–9.

Дмитров В.И., Норенков И.П., Павлов В.В. К проекту Федеральной Программы «Развитие CALS-технологий в России» // Информационные технологии, 1998. №4. С.2–11.

Коновал Д.Г. Повышение эффективности станочных систем на основе разработки регулярных процедур интегрированного автоматизированного проектирования технологического оборудования и систем управления // Юбилейный сборник трудов институтов ОИВТА. М., 1993. Т.1.

Математическое моделирование дискретного производства / Под ред. Ю.М.Соломенцева. М.: ИКТИ РАН, 1993.

Митрофанов В.Г., Калачев О.Н., Схиртладзе А.Г. САПР в технологии машиностроения. Ярославль: Яросл. гос. техн. ун-т, 1995.

Научная школа члена-корреспондента РАН Ю.М.Соломенцева: Сб. науч. трудов. М.: Янус-К, 1999.

Норенков И.П., Кузьмин П.К. Информационная поддержка наукоемких изделий. CALS-технологии. М.: Изд-во МГТУ им. Н.Э.Баумана, 2002.

Павлов В.В. О технологическом мониторинге машиностроительного производства // Техника, Экономика. Сер. Автоматизация проектирования. М.: ВИМИ, 1995. Вып.1–2. С.3–10.

Проблемы CALS-технологий: Сб. науч. трудов / Под ред. В.Г.Митрофанова. М.: Янус-К, 1998.

Ролланд Ф.Д. Основные концепции баз данных. М.: Вильямс, 2002.

Скринский П.Н., Соломенцев Ю.М., Можин Ю.А. Компьютерно-интегрированная система конструкторско-технологической подготовки и управления листоштамповочным и штампоинструментальным производством. М.: Кузнечно-штамповочное производство, 1994. Вып.4.

Соломенцев Ю.М. Конструкторско-технологическая информатика и автоматизация производства. М.: Станкин, 1992.

Соломенцев Ю.М. Проблема создания компьютеризированных интегрированных производств // Автоматизация проектирования. 1997. №1. С.10–14.

Соломенцев Ю.М., Павлов В.В. Моделирование технологической среды машиностроения. М.: Станкин, 1994.

Соломенцев Ю.М., Рыбаков А.В. Компьютерная подготовка производства // Автоматизация проектирования. 1997. №1. С.31–35.

Pavlov V.V. Polychromatic Graphs of Technical System, International Conference «Information Technology in Design EWITD'94». Moscow, 1994. P.117–119.

Zykova S.A., Kolchin A.F. Knowledge-Base Verification in the Process of Modern Intelligent Systems Development and Support // Proceedings of the East-West Conference on Artificial Intelligence, EWAIC'93, Sept.7–9, 1993. (Moscow, Russia). P.99–101.

2.1. Структурное моделирование производственных систем

2.1.1. Структурное моделирование в CALS-технологиях

Особенностью CALS-технологий является необходимость представления информации о всех объектах и процессах, функционирующих на всех этапах жизненного цикла изделий, в едином компьютеризованном информационном пространстве. Информация, формируемая при проектировании, производстве, эксплуатации, техническом обслуживании, ремонте и утилизации изделия, используется и преобразуется при решении разнообразных задач. Многие факторы, влияющие на решение этих задач, плохо формализуемы, и поэтому возможности традиционных математических методов решения часто оказываются недостаточными.

При отсутствии формализованных методов решения некоторых задач компьютеризованного интегрированного производства во всем мире бурно развиваются экспертные системы и унифицированные языковые средства представления используемой информации. Широко известна методология SADT, послужившая основой для создания языка представления данных EXPRESS, регламентированного стандартом ISO 10303. Все большее распространение получает унифицированный язык моделирования UML, в большей мере отражающий специфику описания структуры и процессов функционирования сложных систем. Несмотря на создание указанных средств, проблема разработки методов моделирования CALS-технологий остается весьма актуальной.

Важнейшим компонентом любой автоматизированной системы обработки информации является математическое обеспечение – совокупность математических методов, моделей и алгоритмов, применяемых в автоматизированной системе. Поскольку CALS-технология охватывает все этапы жизненного цикла изделия, математическое обеспечение всех используемых в ней автоматизированных систем обработки информации должно быть интегрированным, взаимосвязанным через единую систему математического моделирования, обеспечивающую:

- создание взаимосвязанных моделей разнородных объектов и процессов (изделий; средств проектирования, производства и эксплуатации; процессов проектирования, производства, управления и т.п.);
- построение взаимосвязанных моделей, соответствующих разному уровню знаний и различной полноте представления данных о моделируемом объекте или процессе;
- прозрачность моделей, смысловую ясность и понятность всех их данных, отношений и связей для пользователя системы;
- адаптацию других методов моделирования, моделей и алгоритмов с целью включения их в единую систему моделирования.

Одной из таких систем, ориентированных на комплексное компьютерное моделирование всех этапов жизненного цикла изделий машиностроения, является разработанная в нашей стране иерархическая система математического моделирования объектов на различных уровнях абстрагирования ИСТРА (Иерархическая система трансляции). Математической основой системы ИСТРА является аппарат полихроматических множеств и полихроматических графов.

Изделия машиностроения и средства обеспечения их жизненного цикла представляют собой сложные технические системы. Одним из важнейших факторов, определяющих эффективность сложной технической системы, является ее структура, характеризующая состав и взаимосвязь включенной в нее компонентов. При структурном моделировании сложных систем широко используется традиционный математический аппарат теории множеств и теории графов. Однако такой аппарат не имеет развитых средств описания разнородных и разнообразных свойств объектов, моделируемых элементами множества или вершинами и ребрами графа, что затрудняет практическое применение данного аппарата при моделировании технических, организационно-экономических, экологических и других реальных систем. Указанный недостаток математического аппарата обычных множеств и графов во многом устраняется при использовании полихроматических множеств и графов.

Свойства, состояние и поведение любой системы взаимосвязаны со свойствами элементов этой системы. В технических системах эти свойства могут иметь различную природу и разнообразные качественные и количественные отношения и связи. Для содержательного моделирования таких систем необходимы средства формализованного представления свойств системы и ее элементов. Такими средствами и обладает математический аппарат теории полихроматических множеств. В отличие от обычного множества, в полихроматическом множестве представляются составы не только абстрактных элементов, но и свойств множества и его элементов. Полихроматический граф представляется полихроматическим множеством вершин и полихроматическим множеством своих ребер или дуг.

При представлении структуры технической системы полихроматическими множествами и полихроматическими графами оказывается возможным формализованное описание состояния и поведения моделируемой системы через описание состава и изменений свойств элементов самой системы в процессе ее функционирования под воздействием внешней среды. Такой аппарат моделирования разрабатывался вначале в связи с созданием автоматизированных систем технической подготовки машиностроительного производства. Структурное моделирование технических систем с использованием аппарата полихроматических множеств и графов было апробировано в различных отраслях машиностроения, и в 80-е годы XX в. отражено в нормативно-технических документах Госстандарта, регламентирующих автоматизированное решение задач конструкторской и технологической подготовки производства.

Современные исследования показывают, что методы математического моделирования технических систем и изделий на всех этапах их жизненного цикла, создаваемые на основе теории полихроматических множеств и графов, органически вписываются в методологию создания CALS-технологий, способствуя повышению эффективности их применения в машиностроении.

2.1.2. Структура полихроматического множества

Согласно канторовскому определению, *множество* S есть любое собрание определенных и различимых между собой объектов нашей интуиции или интеллекта, мыслимое как единое целое. Эти объекты называются *элементами* множества S .

Для представления сложного объекта в виде обычного множества абстрагируются от всех свойств объекта и его составных частей – кроме самого факта существования этого объекта и входящих в него компонентов. При моделировании технических систем наиболее часто применяются конечные счетные множества; такое множество представляет собой совокупность элементов

$$A = (a_1, \dots, a_i, \dots, a_n). \quad (2.1)$$

Для описания того факта, что предметы $(a_1, \dots, a_i, \dots, a_n)$ являются элементами множества, вместо (2.1) можно использовать запись вида

$$A = R^S \{a_1, \dots, a_i, \dots, a_n\}, \quad (2.2)$$

где символ R^S означает, что между перечисленными предметами существует *теоретико-множественное отношение*, наличие которого позволяет рассматривать эти предметы как элементы множества, имя которого обозначено символом A . Если предмет a_i является элементом множества $A - a_i \in A$, то считается, что он обладает свойством принадлежать к данному множеству; в противном случае $a_i \notin A$.

Записи (2.1) и (2.2) отражают один и тот же факт: совокупность предметов, представленных элементами $(a_1, \dots, a_i, \dots, a_n)$, образует качественно новый объект-множество A , представляющее моделируемую систему предметов как единое целое. Поэтому свойства множества A в целом, будучи обусловленными свойствами входящих элементов $a_i \in A$, могут отличаться от свойств этих элементов.

В отличие от обычного множества, в полихроматическом множестве представляются не только составы абстрактных элементов, но и составы свойств множества и его элементов. В теории полихроматических множеств любое свойство самого множества в целом и любое свойство каждого элемента этого множества называется *цветом*. При этом различным свойствам соответствуют различные цвета. Предполагается, что существует *полихроматический универсум ПУ*, включающий в себя все элементы, которые могут быть окрашены в разные цвета, причем один и тот же элемент может иметь одновременно несколько различных цветов. Это позволяет любые объекты, обладающие разнообразными свойствами, представлять элементами универсума ПУ. На практике универсум ПУ реализуется в виде базы данных, включающей описание всех возможных элементов рассматриваемой системы, и всех свойств этих элементов.

Любое свойство реального предмета или явления характеризуется качественной и количественной стороной. Качественная сторона определяет принадлежность рассматриваемого свойства к тому или иному классу качественно различных свойств, а количественная сторона определяет количественную характеристику этого свойства.

Множество, включающее в себя элементы универсума ПУ, называется *полихроматическим множеством ПС*. В полихроматическом множестве и само ПС-множество в целом, и его элементы могут быть окрашены одновременно в несколько разных цветов. Элементы ПС-множества обозначаются как элементы обычного множества A . Каждому элементу $a_i \in A$ соответствует множество цветов

$$F(a_i) = (F_1, \dots, F_j, \dots, F_n), \quad (2.3)$$

а множеству A в целом – множество цветов

$$F(A) = (F_1, \dots, F_j, \dots, F_m). \quad (2.4)$$

Рассматриваемый далее аппарат теории полихроматических множеств основан на следующих допущениях:

- любые два элемента PS -множества будут одноименными, если их полные раскраски одинаковы; такие элементы называются одинаковыми и равными;
- любые два элемента PS -множества имеют разные имена, если их полные раскраски различны; такие элементы не одинаковы и не равны.

Эти допущения являются требованиями к описанию элементов универсума PU . Если при дальнейших исследованиях предметной области состав свойств объекта, отображаемого элементом PS -множества, будет изменен, то это изменение приводится в базе данных универсума PU в соответствии с указанными требованиями.

При моделировании сложного объекта на теоретико-множественном уровне сам объект представляется как PS -множество, а входящие в него компоненты – как элементы этого множества; при этом элементы PS -множества рассматриваются как неделимые объекты. Это не означает, что в реальности они действительно неделимы: большая часть реальных объектов имеет иерархическую структуру, и объект, рассматриваемый как неделимый на одном уровне, расчленяется на составные части при переходе на лежащий ниже уровень иерархии. Поэтому PS -множество рассматривается как выделенная часть иерархической структуры, где данное PS -множество соответствует представлению объекта на определенном уровне иерархии; элементами этого множества будут непосредственно входящие в данный объект компоненты лежащего ниже иерархического уровня.

При решении практических задач исполнитель оперирует объектами предметной области, все свойства которых выявляются при исследовании этой области. Результаты такого исследования и представляются в базе данных универсумом PU элементов, соответствующих объектам данной предметной области. Элементы PU отличаются от элементов обычного множества (2.1) тем, что элементы обычного множества различаются только именами, а элементы PU – именами и составами цветов, соответствующими свойствам объектов, представляемых этими элементами.

Описание раскраски $F(a_i)$ любого элемента a_i , как правило, охватывает не все свойства *депюта*, соответствующего этому элементу, а лишь свойства, известные в данное время или рассматриваемые

в определенных условиях. Так, при решении практических задач часто рассматривают лишь те свойства, которые определяются целью и задачами моделирования.

Составы $F(A)$ цветов PS -множества в целом и цветов $F(a_i)$ его элементов входят в единое множество цветов

$$\mathbf{F} \supseteq F(A), F(a_i); i = 1, 2, \dots, n. \quad (2.5)$$

При математическом моделировании множество цветов (2.5) целесообразно представлять в виде упорядоченного множества. Это позволяет использовать \mathbf{F} как единое булево векторное пространство, в котором раскраски PS -множества и его элементов описываются булевыми векторами.

Первопричиной существования свойств сложного объекта, моделируемого PS -множеством, являются свойства входящих в него элементов. Свойства самих элементов PS -множества, являясь персональными свойствами объектов, отображаемых этими элементами, разделяются на собственные и несобственные, единичные и унитарные; соответствующие этим свойствам цвета также разделяются на собственные и несобственные, единичные и унитарные.

Собственными будут персональные свойства элемента a_i , существующие независимо от наличия или отсутствия взаимосвязи данного элемента с любыми другими элементами, в том числе – независимо от вхождения или невхождения a_i в какое-либо PS -множество. Собственным свойством элемента a_i соответствуют собственные цвета в его раскраске $F(a_i)$.

Свойства элемента a_i , существующие только при наличии взаимосвязи этого элемента с другими элементами, будут *несобственными* свойствами этого элемента. Несобственным свойствам соответствуют *несобственные* цвета в раскраске $F(a_i)$ элемента a_i .

Цвета раскраски $F(A)$ PS -множества также разделяются на собственные и несобственные. *Собственными* будут цвета в $F(A)$, существование которых обусловлено только элементами данного PS -множества. Цвета, обусловленные взаимосвязью данного PS -множества с какими-либо другими объектами вне его, будут несобственными цветами PS -множества.

В раскраске PS -множества или его элемента цвета, существование которых не зависит от других цветов, называются *единичными*, а цвета, существование которых взаимосвязано с существованием других цветов, называются *унитарными*.

Множество $F(a)$ всех различных цветов в персональных раскрасках всех элементов PS -множества определяется соотношением

$$F(a) = \bigcup_{i=1}^n F(a_i). \quad (2.6)$$

Персональные раскраски всех элементов ПIS-множества описываются списком составов $F(a_i)$, $i = 1, 2, \dots, n$, или матрицей бинарных отношений между элементами A и цветами $F(a)$ – булевой матрицей, имеющей вид

$$\|c_{i(j)}\| = [A \times F(a)] = \begin{bmatrix} F_1 & \dots & F_j & \dots & F_m \\ c_{1(1)} & \dots & c_{1(j)} & \dots & c_{1(m)} \\ \dots & \dots & \dots & \dots & \dots \\ c_{i(1)} & \dots & c_{i(j)} & \dots & c_{i(m)} \\ \dots & \dots & \dots & \dots & \dots \\ c_{n(1)} & \dots & c_{n(j)} & \dots & c_{n(m)} \end{bmatrix} \begin{matrix} a_1 \\ \dots \\ a_i \\ \dots \\ a_n \end{matrix} \quad (2.7)$$

В этой матрице истинностное значение элемента $c_{i(j)}$ представляется в виде логической переменной

$$c_{i(j)} = \begin{cases} 1, & \text{если } F_j \in (a_i), \\ 0, & \text{если } F_j \notin (a_i). \end{cases}$$

Любой цвет F_j , как элемент множества (2.3) или (2.4), может быть представлен логической переменной в раскраске $F(a_i)$

$$F_j = \begin{cases} 1, & \text{если } F_j \in (a_i), \\ 0, & \text{если } F_j \notin (a_i). \end{cases}$$

или логической переменной в раскраске $F(A)$

$$F_j = \begin{cases} 1, & \text{если } F_j \in (A), \\ 0, & \text{если } F_j \notin (A). \end{cases}$$

Это позволяет записать соотношение (2.6) между множествами $F(a)$ и $F(a_i)$ в виде логической формулы

$$F(a) = \bigvee_{i=1}^n F(a_i), \quad (2.8)$$

более удобной при компьютерном моделировании, где $F(a)$ и $F(a_i)$ представлены как булевы векторы в едином булевом векторном пространстве с составом цветов (2.5). Персональная раскраска $F(a_i)$ элемента a_i определяется составом элементов $c_{i(j)} = 1$ i -й строки матрицы (2.7). Описание цвета F_j как логической переменной или теоретико-множественной величины позволяет переходить от одной формы представления к другой, или использовать обе формы одновременно. Совместное использование теоретико-множественного и логического аппарата широко распространено в литературе.

Логические операции над раскрасками в PS -множествах удобно выполнять как операции над булевыми векторами в булевом векторном пространстве, соответствующем единственному множеству цветов (2.5).

Существование цвета $F_j(A)$ PS -множества обусловлено существованием одноименного персонального цвета $F_j(a_i)$ у элемента $a_i \in A$ или у нескольких таких элементов. Однако, учитывая особенности моделирования реальных объектов и систем, допускается обусловленность существования $F_j(A)$ в зависимости от любых других, а не только одноименных цветов $a_i \in A$. Поэтому, если в составе цветов элемента a_i , влияющего на существование унитарного цвета $F_j(A)$, нет одноименного цвета, то такой цвет $F_j(a_i)$ формально вводится в состав $F(a_i)$ и устанавливается его взаимосвязь с другими персональными цветами a_i , в действительности влияющими на существование $F_j(A)$. В результате состав унитарных цветов $F(A)$ соотносится с составом $F(a)$ персональных цветов всех элементов как его подмножество

$$F(A) \subseteq F(a). \quad (2.9)$$

Указанный прием позволяет обобщенно описывать взаимосвязь унитарной раскраски PS -множества и раскрасок элементов $a_i \in A$ в булевой матрице

$$\|c_{i(j)}\| = [A \times F(A)], \quad (2.10)$$

с отражением в ней только унитарных цветов $F(A)$ и одноименных с ними цветов $F(a_i)$. В матрице (2.10) значение $c_{i(j)} = 1$, если $F_j \in F(a_i)$, причем $F_j(a_i)$ влияет на существование $F_j(A)$, и $c_{i(j)} = 0$ – в противном случае. Состав всех элементов $a_i \in A$, одноименные цвета которых влияют на существование унитарного цвета $F_j(A)$, определяется составом равных единице элементов j -го столбца этой матрицы. Матрица (2.10) отличается тем, что $F(A)$ содержит только унитарные цвета и одноименные с ними персональные цвета элементов, влияющие на существование унитарных цветов, в то время как $F(a)$ содержит все различные цвета всех элементов $a_i \in A$. Состав

$$A_k(F_j) = \{a_{i_1}, a_{i_2}, \dots, a_{i_n}\} \quad (2.11)$$

элементов $a_i \in A$, при наличии которых унитарный цвет $F_j(A)$ существует, называется *телом* данного унитарного цвета. Если существует n вариантов тел унитарного цвета $F_j(A)$, то общий состав элементов PS -множества, влияющих на существование данного унитарного цвета, равен

$$A(F_j) = \bigcup_{k=1}^n A_k(F_j). \quad (2.12)$$

Приняв символы элементов $a_i \in A$ как логические переменные

$$a_i = \begin{cases} 1, & \text{если } a_i \in A_k(F_j), \\ 0, & \text{если } a_i \notin A_k(F_j), \end{cases}$$

состав (2.11) элементов тела $A_k(F_j)$ записывается в виде булева вектора

$$A_k(F_j) = \{a_{i_1}, a_{i_2}, \dots, a_{i_n}\},$$

а общий состав элементов, влияющих на существование $F_j(A)$, определяется логическим уравнением

$$A(F_j) = \bigvee_{k=1}^n A_k(F_j). \quad (2.13)$$

Составы элементов всех вариантов тел, обеспечивающих существование всех унитарных цветов $F(A)$ *ПС*-множества, описываются булевой матрицей

$$\|c_{i(j)}\| = [A \times A(F)], \quad (2.14)$$

где $A(F)$ есть объединение всех вариантов тел унитарных цветов $F(A)$, или, с учетом (2.13),

$$A(F) = \bigvee_{i=1}^m A(F_j) = \bigvee_{i=1}^m \bigvee_{k=1}^n A_k(F_j). \quad (2.15)$$

Удобной формой является представление *ПС*-множества шестеркой компонентов

$$PS = (A, F(a), F(A), [A \times F(a)], [A \times F(A)], [A \times A(F)]). \quad (2.16)$$

Здесь *ПС*-множества компоненты A , $F(a)$ и $F(A)$, как наиболее часто рассматриваемые отдельно при решении прикладных задач, представлены в явном виде, а компоненты $F(a_i)$, $A_k(F_j)$, $A(F_j)$ и $A(F)$ – в составах булевых матриц $[A \times F(a)]$, $[A \times F(A)]$ и $[A \times A(F)]$.

Такое представление *ПС*-множества будет адекватным моделируемому объекту только в случае, если можно не учитывать взаимосвязь между цветами *ПС*-множества и цветами его элементов по условиям их существования. При учете такой взаимосвязи в описание *ПС*-множества должны вводиться, в дополнение к составу компонентов (2.16), теоретико-множественные и логические отношения, отражающие эти связи. Введение таких отношений существенно расширяет моделирующие возможности *ПС*-множества за счет описания причинно-следственных связей между элементами A и цветами $F(a)$ и $F(A)$. Добавление таких отношений превращает *ПС*-множество в единую, логически связанную математическую структуру, способную

отобразить широкий спектр структурных свойств моделируемой сложной системы.

В практической деятельности существенное значение имеет степень соответствия свойств моделируемого объекта желаемым или заранее заданным значениям. Поэтому одними из главных факторов, определяющих свойства ПС-множества, являются условия существования его цветов и раскрасок в целом.

В тех случаях, когда условие существования цвета F_j определяется множеством параметров M_j , имеющих числовые значения, истинностные значения F_j как логической переменной определяются из соотношений

$$F_j = \begin{cases} 1, & \text{если } \forall m_k \in M_j (\omega_k \subseteq \Delta_k), \\ 0, & \text{если } \exists m_k \in M_j (\omega_k \setminus \Delta_k \neq \emptyset), \end{cases} \quad (2.17)$$

где ω_k – поле рассеяния погрешностей величины параметра m_k ; Δ_k – поле допуска на погрешности величины параметра m_k .

Если на существование собственного цвета $F_j(a_i)$ влияют другие цвета в $F(a_i)$, то этот цвет будет унитарным. Если унитарный цвет $F_j(a_i)$ существует при наличии различных вариантов составов других цветов $F(a_i)$, то такие составы определяются логическим уравнением

$$F_j(a_i) = R^L(F_{k_1}(a_i), F_{k_2}(a_i), \dots, F_{k_m}(a_i)), \quad (2.18)$$

где цвета $F_k(a_i)$ представляются как упорядоченные логические переменные, имеющие соответствующие истинностные значения.

Если после подстановки истинностных значений влияющих цветов в уравнении (2.18) значение $F_j(a_i)$ будет истинно, то этот цвет в раскраске $F(a_i)$ существует:

$$F_j(a_i) = 1, \text{ если } F_j(a_i) = R^L(F_{k_1}(a_i), F_{k_2}(a_i), \dots, F_{k_m}(a_i)) = 1. \quad (2.19)$$

Реальные условия, определяющие зависимость существования $F_j(a_i)$ от других цветов, могут быть различными. Такие различия обусловлены природой рассматриваемых факторов, интенсивностью их проявления, степенью абстрагирования и т.п. Эта зависимость может выражаться либо только в качественной форме, либо в качественной и количественной форме.

Взаимосвязь всех цветов $F(a_i)$ по условиям их существования может быть представлена единым математическим объектом как подмножество декартова произведения $F(a_i) \times F(a_i)$ в виде булевой матрицы

$$\|c_{k(j)}\| = [F(a_i) \times T(a_j)] = \begin{bmatrix} F_1 & \dots & F_j & \dots & F_m \\ c_{1(1)} & \dots & c_{j(1)} & \dots & c_{1(m)} \\ \dots & \dots & \dots & \dots & \dots \\ c_{k(1)} & \dots & c_{k(j)} & \dots & c_{k(m)} \\ \dots & \dots & \dots & \dots & \dots \\ c_{m(1)} & \dots & c_{m(j)} & \dots & c_{m(n)} \end{bmatrix} \begin{matrix} F_1 \\ F_k \\ F_m \end{matrix}, \quad (2.20)$$

где $c_{k(j)} = 1$ – если существование $F_j(a_i)$ зависит от $F_k(a_i)$. Если $F_j(a_i)$ существует независимо от других цветов, то диагональный элемент $c_{j(j)} = 1$.

Булева матрица (2.20) описывает наличие взаимосвязи цветов по условиям их существования в $F(a_i)$. Свойство, характеризующее наличие такой взаимосвязи между цветами $F_j(a_i)$ и $F_k(a_i)$, отображается элементом $c_{j(k)} = 1$ и/или $c_{k(j)} = 1$ этой матрицы. В случае, когда существование одного F_k недостаточно для существования F_j , наличие F_k *влияет*, но не обеспечивает существования F_j . В этом случае существование F_j *обеспечивает* только группа одновременно существующих цветов $\{F_{k_1}, F_{k_2}, \dots, F_{k_n}\}$.

Если полный состав цветов, влияющих на существование F_j в $F(a_i)$, неоднороден по своему влиянию на F_j , то в этом случае состав влияющих цветов можно разбить на m групп, таких, что при одновременном наличии всех цветов F_{k_p} данной группы цвет $F_j(a_i)$ существует. В этом случае зависимость (2.18) существования $F_j(a_i)$ от других цветов в $F(a_i)$ описывается логическим уравнением дизъюнктивной нормальной формы, сокращенно – ДНФ:

$$F_j(a_i) = \bigvee_{q=1}^m \bigwedge_{p=1}^{n_q} F_{k_p}(a_i)_q. \quad (2.21)$$

Цвет F_{k_p} здесь представляется как логическая переменная $F_{k_p} = 1$, если $F_{k_p} \in F(a_i)$, и $F_{k_p} = 0$ – в противном случае. При наличии в матрице (2.20) элемента $c_{j(j)} = 1$ в правую часть уравнения (2.21) включается и сам цвет $F_j(a_i)$ в качестве отдельной группы, состоящей из одного цвета $F_j(a_i) = 1$.

Иногда более удобно зависимость (2.18) существования унитарного цвета $F_j(a_i)$ от других цветов в $F(a_i)$ описывать логическим уравнением конъюнктивной нормальной формы (КНФ)

$$F_j(a_i) = \bigwedge_{q=1}^m \bigvee_{p=1}^{n_q} F_{k_p}(a_i)_q. \quad (2.22)$$

Смысловое содержание этой формулы интерпретируется как зависимость существования $F_j(a_i)$ от одновременного существования

не менее чем одного цвета из каждой q -й группы дизъюнктивно-связанных цветов, поскольку все n_q цветов, входящих в такую группу, рассматриваются существующими независимо друг от друга. Состав всех цветов, входящих в правую часть уравнения (2.22) и влияющих на существование $F_j(a_i)$, как и при ДНФ, отображается в булевой матрице (2.20), и соответствует составу ее элементов $c_{k(j)} = 1$ в j -м столбце матрицы.

Моделирование на уровне теоретико-множественных и логических отношений не заменяет и не подменяет моделирование количественных отношений между свойствами моделируемого объекта, а отражает лишь структурные и причинно-следственные связи между этими свойствами по условиям их существования, что оказывается полезным при построении баз данных и структурном моделировании сложных объектов. Условие (2.19) существования $F_j(a_i)$ в $F(a_i)$, исходя из (2.21), определяется соотношением

$$F_j(a_i) = 1, \text{ если } F_j(a_i) = \bigvee_{p=1}^m \bigwedge_{q=1}^{n_q} F_{k_p}(a_i)_q = 1. \quad (2.23)$$

При описании зависимости (2.18) уравнением КНФ вида (2.22) условие (2.19) существования $F_j(a_i)$ в $F(a_i)$ определяется соотношением

$$F_j(a_i) = 1, \text{ если } F_j(a_i) = \bigvee_{q=1}^m \bigwedge_{p=1}^{n_q} F_{k_p}(a_i)_q = 1. \quad (2.24)$$

Несобственные цвета элемента a_i являются унитарными, поскольку их существование зависит от других элементов PS -множества. Если цвета элемента a_i зависят по условиям их существования и от других собственных свойств, и от цветов другого элемента a_j , то такая зависимость описывается блочной булевой матрицей

$$\begin{array}{cc} & F(a_i) \\ [F(a_j), (F(a_i) \times F(a_i))] = [F(a_j) \times F(a_i)] & F(a_j) \\ & [F(a_i) \times F(a_i)] & F(a_i) \end{array} \quad (2.25)$$

где блок $[F(a_i) \times F(a_i)]$ суть матрица вида (2.20), а блок $[F(a_j) \times F(a_i)]$ отражает влияние цветов $F(a_j)$ на цвета $F(a_i)$.

Если существование несобственных цветов элемента a_i связано с существованием цветов нескольких других элементов $(a_{j_1}, \dots, a_{j_p}, \dots, a_{j_n})$ PS -множества, то состав таких связей описывается булевой матрицей

$$[(F(a_{j_1}), \dots, F(a_{j_p}), \dots, F(a_{j_n})) \times F(a_i)] =$$

$$\begin{aligned}
 & F(a_i) \\
 = & \begin{bmatrix} F(a_{j_1}) \times F(a_i) \\ \dots \\ F(a_{j_p}) \times F(a_i) \\ \dots \\ F(a_{j_n}) \times F(a_i) \end{bmatrix} \begin{matrix} F(a_{j_1}) \\ \dots \\ F(a_{j_p}) \\ \dots \\ F(a_{j_n}) \end{matrix}
 \end{aligned} \tag{2.26}$$

В этом случае зависимость существования унитарного цвета $F_{j(a_i)}$ элемента a_j от существования разных цветов других элементов PS -множества определяется уравнением

$$F_j(a_i) = R^L(F_{k_1}(a_{j_1}), F_{k_2}(a_{j_1}), \dots, F_{k_m}(a_{j_n})), \tag{2.27}$$

состав правой части которого формируется, исходя из истинностных значений элементов $c_{k(j)}$ в j -ом столбце булевой матрицы (2.26). Условие существования цвета $F_j(a_i)$ в этом случае имеет вид

$$\begin{aligned}
 F_j(a_i) = 1, \text{ если } F_j(a_i) = R^L(F_{k_1}(a_{j_1}), \\
 F_{k_2}(a_{j_1}), \dots, F_{k_m}(a_{j_n})) = 1.
 \end{aligned} \tag{2.28}$$

Логическое уравнение (2.27) по своей структуре аналогично уравнению (2.18) и может быть представлено уравнением ДНФ или КНФ.

Раскраска $F(a_i)$ характеризует свойства элемента a_i в целом. Между раскраской $F(a_i)$, как единым целым, и входящими в нее цветами существуют отношения, которые формально можно представить, с учетом логических отношений между цветами $F(a_i)$, логическими уравнениями вида

$$F(a_i) = R^L(F_{j_1}(a_i), F_{j_2}(a_i), \dots, F_{j_n}(a_i)). \tag{2.29}$$

Соответствие раскраски элемента a_i ее смысловой интерпретации представляется как истинность логического уравнения (2.29). Поэтому соответствие $F(a_i)$ требуемым условиям определяется соотношением

$$F(a_i) = 1, \text{ если } F(a_i) = R^L(F_{j_1}(a_i), F_{j_2}(a_i), \dots, F_{j_n}(a_i)) = 1. \tag{2.30}$$

Описание полного состава цветов a_i осуществляется либо теоретико-множественным отношением в виде списка цветов, входящих в множество $F(a_i)$, либо логическим уравнением (2.29) вида

$$F(a_i) = F_1(a_i) \wedge F_2(a_i) \wedge \dots \wedge F_n(a_i) = \bigwedge_{p=1}^n F_{j_p}(a_i), \tag{2.31}$$

поскольку в составе раскраски $F(a_i)$ должны присутствовать все цвета: и $F_1(a_i)$, и $F_2(a_i)$, и т.д., вплоть до $F_n(a_i)$. Наличие полного состава цветов $F(a_i)$, присущих элементу a_i , в этом случае определяется соотношением

$$F(a_i) = 1, \text{ если} \\ F(a_i) = F_1(a_i) \wedge F_2(a_i) \wedge \dots \wedge F_n(a_i) = \bigwedge_{p=1}^n F_{j_p}(a_i) = 1. \quad (2.32)$$

Описание возможных вариантов составов цветов a_i осуществляется либо списками таких составов, либо логическим уравнением (2.29) в ДНФ

$$F(a_i) = \bigvee_{q=1}^m \bigwedge_{p=1}^{n_q} F_{j_p}(a_i)_q, \quad (2.33)$$

где m – число конъюнктивно-связанных групп цветов, соответствующих возможным составам одновременно существующих цветов в раскраске $F(a_i)$; n_q – число цветов в группе с индексом q . В этом случае соответствие состава $F(a_i)$ цветов элемента a_i заданным условиям определяется соотношением

$$F(a_i) = 1, \text{ если} \quad F(a_i) = \bigvee_{q=1}^m \bigwedge_{p=1}^{n_q} F_{j_p}(a_i)_q = 1. \quad (2.34)$$

Уравнение (2.29) может быть представлено не только в ДНФ, как уравнение (2.33), но и в КНФ:

$$F(a_i) = \bigwedge_{q=1}^m \bigvee_{p=1}^{n_q} F_{j_p}(a_i)_q. \quad (2.35)$$

В этом случае соответствие состава $F(a_i)$ цветов элемента a_i заданным условиям определяется соотношением

$$F(a_i) = 1, \text{ если} \quad F(a_i) = \bigwedge_{q=1}^m \bigvee_{p=1}^{n_q} F_{j_p}(a_i)_q = 1. \quad (2.36)$$

Объект моделирования, рассматриваемый как элемент PS -множества, может быть представлен на разных уровнях детализации его описания. Описания элемента a_i на различных уровнях детализации могут быть такими:

$$\langle a_i, F(a_i) \rangle, \quad (2.37)$$

где a_i – где имя элемента, $F(a_i)$ – множество цветов этого элемента;

$$\langle a_i, F(a_i), [F(a_i) \times F(a_i)] \rangle, \quad (2.38)$$

$$\langle a_i, F(a_i), [F(a_i) \times F(a_i)], \{F_j(a_i)_{(2.18)}, j = 1, 2, \dots, n\} \rangle, \quad (2.39)$$

где $F_j(a_i)_{(2.18)}$ – логическое уравнение вида (2.21) или (2.22), и т.д.

Персональная раскраска $F(A)$ ПС-множества отображает свойства сложного объекта, составные части которого представляются элементами этого множества. В общем случае на существование унитарного цвета $F_j(A)$ могут влиять как одноименные с ним цвета $F_j(a_i)$ элементов $a_i \in A$, так и другие цвета этих элементов. Влияние цветов элементов на существование $F_j(A)$ с учетом логических связей между цветами элементов удобно описывать логическим уравнением

$$F_j(A) = R^L(F_k(a_i); i = 1, 2, \dots, n; k = 1, 2, \dots, m). \quad (2.40)$$

Цвет элемента $F_k(a_i)$ входит в правую часть уравнения (2.40) как логическая переменная, истинностные значения которой зависят от выполнения условия (2.19) или (2.28) существования этого цвета в раскраске $F(a_i)$. В простейшем случае существование каждого унитарного цвета $F_j(A)$ обусловлено наличием одноименных цветов в раскрасках элементов ПС-множества. В этом случае вместо отношения (2.40) рассматривается отношение

$$F_j(A) = R^L(F_j(a_i); i = 1, 2, \dots, n). \quad (2.41)$$

Состав унитарных цветов $F(A)$ при этом определяется с помощью булевой матрицы (2.10). В этой матрице определяются составы элементов вида (2.11), обладающих персональными цветами $F_j(a_{i_p})$, и образующих тела $A_k(F_j)$ унитарного цвета $F_j(A)$. Отношение (2.41), определяющее взаимосвязь унитарного цвета $F_j(A)$ и одноименных цветов $F_j(a_{i_p})$ по условиям их существования, описывается логическим уравнением дизъюнктивной нормальной формы (ДНФ)

$$F_j(A) = \bigvee_{k=1}^m \bigwedge_{p=1}^{n_k} F_j(a_{i_p})_k, \quad (2.42)$$

где m – количество вариантов тел унитарного цвета $F_j(A)$. Взаимосвязь $F_j(A)$ с одноименными цветами элементов ПС-множества по условиям существования может быть представлена и в виде логического уравнения конъюнктивной нормальной формы (КНФ)

$$F_j(A) = \bigwedge_{k=1}^m \bigvee_{p=1}^{n_k} F_j(a_{i_p})_k. \quad (2.43)$$

Однако уравнение ДНФ (2.42) удобнее тем, что оно отображает в явном виде составы элементов, образующих варианты $A_k(F_j)$ тел унитарного цвета $F_j(A)$. Условие существования унитарного цвета $F_j(A)$ в зависимости от существования одноименных цветов элементов ПС-множества в случае (2.41) определяется соотношением

$$F_j(A) = 1, \text{ если } F_j(A) = R^L(F_j(a_i); i = 1, 2, \dots, n) = 1. \quad (2.44)$$

После определения состава $F(A)$ унитарных цветов, для которых выполняется условие их существования, формируется булева матрица (2.10) взаимосвязи унитарных цветов с одноименными персональными цветами элементов, входящих в состав тел вида (2.11). Цвета элементов, не образующих ни одного тела вида (2.11), не порождают одноименных им унитарных цветов; в этом случае состав $F(A)$ унитарных цветов оказывается меньше состава $F(A)$.

Если $F_j(A)$ существует только при одновременном наличии какой-либо группы $\{F_{k_1}, F_{k_2}, \dots, F_{k_n}\}$ цветов a_i , то уравнение (2.40) описывается как логическое уравнение ДНФ

$$F_j(A) = \bigvee_{q=1}^m \bigwedge_{p=1}^{n_q} F_{k_p}(a_i)_q, \quad (2.45)$$

где m – количество таких групп цветов элемента a_i . Условие (2.44) существования унитарного цвета $F_j(A)$ при этом определяется соотношением

$$F_j(A) = 1, \text{ если } F_j(A) = \bigvee_{q=1}^m \bigwedge_{p=1}^{n_q} F_{k_p}(a_i)_q = 1. \quad (2.46)$$

Если зависимость существования $F_j(A)$ от цветов элемента a_i описывается логическим уравнением КНФ

$$F_j(A) = \bigwedge_{q=1}^m \bigvee_{p=1}^{n_q} F_{k_p}(a_i)_q, \quad (2.47)$$

то условие существования $F_j(A)$ определяется соотношением

$$F_j(A) = 1, \text{ если } F_j(A) = \bigwedge_{q=1}^m \bigvee_{p=1}^{n_q} F_{k_p}(a_i)_q = 1. \quad (2.48)$$

Взаимосвязь персональных цветов всех элементов $ПС$ -множества по условиям их существования представляется в блочной булевой матрице

$$[(F(a_1), \dots, F(a_i), \dots, F(a_n)) \times (F(a_1), \dots, F(a_j), \dots, F(a_n))), \quad (2.49)$$

где для каждой пары элементов $a_i, a_j \in A$, персональные цвета которых взаимосвязаны по условиям их существования, имеются блоки, соответствующие блокам матрицы (2.25). Для учета влияния цветов всех n элементов $ПС$ -множества на его унитарные цвета строится единая блочная матрица

$$[(F(a_1), F(a_2), \dots, F(a_n)) \times (F(A), F(a_1), F(a_2), \dots, F(a_n))), \quad (2.50)$$

по которой можно определить состав всех цветов различных элементов PS -множества, влияющих на существование унитарных цветов в $F(A)$. Если влияние цветов элементов на существование унитарных цветов в $F(A)$ отображено во взаимосвязи только одноименных цветов элементов и унитарных цветов, то для описания состава влияющих на цвета в $F(A)$ цветов элементов целесообразно использовать столбцовую матрицу

$$[(F(a_1), F(a_2), \dots, F(a_n)) \times F(A)] = \begin{matrix} & F(A) \\ \begin{bmatrix} [F(a_1) \times F(A)] \\ [F(a_2) \times F(A)] \\ \dots \\ [F(a_n) \times F(A)] \end{bmatrix} & \begin{matrix} F(a_1) \\ F(a_2) \\ \dots \\ F(a_n) \end{matrix} \end{matrix} \quad (2.51)$$

являющуюся первым столбцом блочной матрицы (2.50). Состав

$$\{c_{i_1(j)} = 1, c_{i_2(j)} = 1, \dots, c_{i_n(j)} = 1\}$$

равных единице элементов j -го столбца матрицы (2.51) определяет состав

$$A(F_j) = \{a_{i_1}, a_{i_2}, \dots, a_{i_n}\}$$

элементов PS -множества, одноименные цвета $F_j(a_{i_p})$ которых влияют на существование унитарного цвета $F_j(A)$. Из этих элементов формируются составы (2.11) тел $A_k(F_j)$, реализующих унитарный цвет $F_j(A)$.

Хотя первопричиной существования унитарных цветов PS -множества являются цвета его элементов, но в тех случаях, когда PS -множество выступает как самостоятельный единый объект, целесообразно рассматривать взаимосвязь между самими цветами в раскраске $F(A)$. В отрыве от раскрасок элементов цвета PS -множества разделяются на собственные и несобственные по отношению к другим PS -множествам, и на единичные и унитарные внутри $F(A)$. Существование единичного цвета $F_j(A)$ не зависит от других цветов в $F(A)$; наоборот, существование унитарного цвета $F_j(A)$ зависит от существования других цветов в $F(A)$.

Рассматриваются следующие виды взаимосвязи цветов $F(A)$:

– взаимосвязь отдельного унитарного цвета $F_j(A)$ с другими цветами $F(A)$, описываемая логическим уравнением

$$F_j(A) = R^L(F_{j_1}(A), F_{j_2}(A), \dots, F_{j_m}(A)). \quad (2.52)$$

– взаимосвязь раскраски $F(A)$ в целом с входящими в нее цветами, описываемая логическим уравнением

$$F(A) = R^L (F_1(A), F_2(A), \dots, F_m(A)). \quad (2.53)$$

При описании условий существования унитарного цвета $F_j(A)$ устанавливают взаимосвязь $F_j(A)$ с другими цветами, влияющими на его существование в раскраске $F(A)$. Составы цветов, влияющих на существование каждого отдельного унитарного цвета, определяются булевой матрицей

$$\| \| c_{k(j)} \| \| = [F(A) \times F(A)]. \quad (2.54)$$

Зависимость (2.52) существования $F_j(A)$ от других унитарных цветов в $F(A)$ по матрице (2.54) определяется так же как зависимость существования $F_j(a_i)$ по матрице (2.20). Состав цветов, влияющих на существование $F_j(A)$, определяет j -й столбец матрицы (2.54); более детально влияние этих цветов на существование $F_j(A)$ определяется соотношением

$$F_j(A) = 1, \text{ если } F_j(A) = R^L (F_{j_1}(A), F_{j_2}(A), \dots, F_{j_m}(A)) = 1. \quad (2.55)$$

Уравнение (2.52) может быть представлено как уравнение ДНФ

$$F_j(A) = \bigvee_{q=1}^m \bigwedge_{p=1}^{n_q} F_{j_p}(A)_q, \quad (2.56)$$

где $F_{k_p}(A) = 1$, если F_{k_p} существует, т.е. $F_{k_p} \in F(A)$, и $F_{k_p} = 0$ – в противном случае. Наличие элемента $c_{j(j)} = 1$ в булевой матрице (2.54) свидетельствует о том, что $F_j(A)$ может существовать независимо от других унитарных цветов; в этом случае $F_j(A)$ включается в правую часть уравнения (2.56), как это делается в уравнении (2.21). Условие (2.55) существования $F_j(A)$ в $F(A)$ в зависимости от существования других унитарных цветов в этом случае определяется соотношением

$$F_j(A) = 1, \text{ если } F_j(A) = \bigvee_{q=1}^m \bigwedge_{p=1}^{n_q} F_{j_p}(A)_q = 1. \quad (2.57)$$

Взаимосвязь $F_j(A)$ с другими унитарными цветами может быть представлена и логическим уравнением КНФ

$$F_j(A) = \bigwedge_{q=1}^m \bigvee_{p=1}^{n_q} F_{j_p}(A)_q; \quad (2.58)$$

в этом случае условие существования $F_j(A)$ описывается соотношением

$$F_j(A) = 1, \text{ если } F_j(A) = \bigwedge_{q=1}^m \bigvee_{p=1}^{n_q} F_{j_p}(A)_q = 1. \quad (2.59)$$

Отношение (2.53) между унитарной раскраской $F(A)$ как единым целым и входящими в нее цветами также может быть представлено как логическое уравнение дизъюнктивной или конъюнктивной нормальной формы. Условие существования или соответствия унитарной раскраски $F(A)$ заданным требованиям в зависимости от состава входящих цветов определяется соотношением

$$F(A) = 1, \text{ если } F(A) = R^L(F_1(A), F_2(A), \dots, F_m(A)) = 1. \quad (2.60)$$

Наличие всех связей между унитарными цветами $F(A)$ и цветами $F(a_i)$ всех элементов PS -множества описывается единой блочной булевой матрицей

$$[(F(A), F(a_1), F(a_2), \dots, F(a_n)) \times (F(A), F(a_1), F(a_2), \dots, F(a_n)))], \quad (2.61)$$

частями которой являются булевы матрицы (2.20), (2.25), (2.26) и другие. Матрица (2.61) обеспечивает системную связность всех других булевых матриц, являющихся частями этой матрицы.

Исходя из условий решаемой задачи, свойства объекта, моделируемого PS -множеством, целесообразно рассматривать на разных уровнях детализации их представления. Это достигается за счет разных уровней детализации описания взаимосвязи и условий существования унитарных цветов PS -множества и цветов его элементов. Составы элементов и цветов PS -множества могут быть описаны на следующих уровнях детализации:

$$\langle A, F(A) \rangle, \quad (2.62)$$

$$\langle A, F(A), [A \times F(A)] \rangle, \quad (2.63)$$

$$\langle A, F(A), [A \times F(A)], [A \times F(A)], [A \times A(F)] \rangle, \quad (2.64)$$

$$\langle A, F(a), F(A), [A \times F(a)], [A \times F(A)] \rangle, \quad (2.65)$$

$$\langle A, F(a), F(A), [A \times F(a)], [A \times F(A)], [A \times A(F)] \rangle \quad (2.66)$$

Представление PS -множества шестеркой компонентов (2.16) является наиболее детальным уровнем описания состава элементов и цветов этого множества, соответствующим уровню детализации (2.66).

Моделирующие возможности PS -множества существенно зависят от особенностей тел (2.11), отображающих влияние персональных цветов элементов PS -множества на существование его унитарных цветов.

Если телом любого унитарного цвета $F_j \in F(A)$ является каждый элемент $a_i \in A$, имеющий персональный цвет $F_j(a_i)$, и каждому персональному цвету любого элемента a_i соответствует одноименный

унитарный цвет PS -множества, то взаимосвязь унитарных и персональных цветов в PS -множестве называется *дизъюнктивной формой связи* (ДФС), а само PS -множество – *дизъюнктивным* и обозначается символом $\Pi^{\vee}S$. Если в $\Pi^{\vee}S$ -множестве имеется m элементов с персональным цветом F_j , то каждый такой элемент является телом $F_j(A)$, поэтому уравнение (2.13) имеет вид

$$A(F_j) = \bigvee_{p=1}^m a_{i_p}, \quad \forall a_{i_p} (F_j \in F(a_{i_p})). \quad (2.67)$$

Отсюда следует, что при ДФС унитарный цвет $F_j(A)$ существует при наличии хотя бы одного элемента $a_i \in A$ с персональным цветом $F_j(a_i)$.

Унитарная раскраска $F(A)$ и персональные раскраски элементов $\Pi^{\vee}S$ -множества связаны логическим отношением

$$F(A) = F(a) = \bigvee_{i=1}^n F(a_i), \quad (2.68)$$

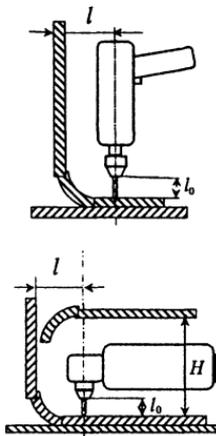
В $\Pi^{\vee}S$ -множестве персональный цвет $F_j(a_i)$ любого элемента *порождает* унитарный цвет $F_j(A)$ и a_i является телом этого цвета; одновременно этот элемент является телом и всех других унитарных цветов, одноименных остальным персональным цветам $F(a_i)$. Поэтому состав элементов и цветов $\Pi^{\vee}S$ -множества достаточно полно описывается на уровне (2.63) и имеет вид

$$\Pi^{\vee}S = (A, F(A), [A \times F(A)]). \quad (2.69)$$

PS -множество можно рассматривать как своеобразную форму представления реляционной модели данных, в которой отношение между сущностями и их атрибутами описывается логическими переменными в виде бинарного отношения. Например, реляционная модель (рис. 2.1а) представляется булевой матрицей (рис. 2.1б) PS -множества вида (2.69) с составом элементов $A = (a_1, a_2, \dots, a_6)$ и составом унитарных цветов $F(A) = (F_1, F_2, \dots, F_{15})$.

Если отношение (2.67) для всех цветов $F_j \in F(A)$ не выполняется, то PS -множество не может быть дизъюнктивным. В этом случае тело (2.11) унитарного цвета $F_j(A)$ содержит более одного элемента, и для независимого от других цветов $F(A)$ отдельного цвета $F_j(A)$ определяется формулой

$$A_k(F_j) = \bigwedge_{q=1}^{m_k} a_{i_q}, \quad \forall a_{i_q} (F_j \in F(a_{i_q})), \quad (2.70)$$



а)

| | | Материал детали | Наибольший диаметр отверстия | | |
|--------------|-------|-----------------|---------------------------------|-----------------------------|-----|
| | | | Минимальное расстояние l , мм | Минимальная высота H , мм | |
| СМ-3-18000 | a_1 | Ал. | 3 | 34 | 600 |
| СМ21-6-12000 | a_2 | Ал. | 6 | 34 | 600 |
| СМ21-9-300 | a_3 | Ст.Тит | 9 | 34 | 600 |
| СМ-21-9-2500 | a_4 | Ал. | 9 | 34 | 600 |
| СМУ11-6-1700 | a_5 | Ст.Тит | 6 | 17 | 68 |
| УСМ21-20М | a_6 | Ал. | 8 | 19 | 34 |

б)

| | Ал. сплавы | | Ст./Тит. сплавы | | | | | | | | | | | | |
|--------------|------------|-------|----------------------------------|---|---|---|---------------------------------|----|----|----|-----------------------------|----|-----|---|---|
| | F_1 | F_2 | Наибольший диаметр отверстия, мм | | | | Минимальное расстояние l , мм | | | | Минимальная высота H , мм | | | | |
| | | | 3 | 6 | 8 | 9 | 17 | 19 | 24 | 34 | 34 | 68 | 600 | | |
| СМ-3-18000 | a_1 | ● | ● | | | | | | | | ● | ● | | | ● |
| СМ21-6-12000 | a_2 | ● | ● | ● | ● | | | | | | | ● | | | ● |
| СМ21-9-300 | a_3 | | ● | ● | ● | ● | | | | | | ● | | | ● |
| СМ-21-9-2500 | a_4 | ● | | ● | ● | | | | | | | ● | | | ● |
| СМУ11-6-1700 | a_5 | | ● | ● | ● | | | ● | ● | ● | ● | | | ● | ● |
| УСМ21-20М | a_6 | ● | | ● | ● | ● | | | ● | ● | ● | ● | ● | ● | ● |

Рис.2.1. Описание свойств сверлильных машин

а) реляционная модель свойств: Ал – алюминиевые сплавы, Ст – стали, Тит – титановые сплавы б) булева матрица свойств

Такая взаимосвязь унитарных цветов $F(A)$ и персональных цветов элементов PS -множества называется *конъюнктивной формой связи* (КФС), а само полихроматическое множество также называется *конъюнктивным* и обозначается символом $P \wedge S$. При наличии тела вида (2.70) условие существования отдельного цвета $F_j(A)$ имеет вид

$$F_j(A) = 1, \text{ если } A_k(F_j) = \bigwedge_{q=1}^m a_{i_q} = 1, \forall a_{i_q} (F_j \in F(a_{i_q})) \quad (2.71)$$

или

$$F_j(A) = 1, \text{ если } F_j(A) = \bigwedge_{q=1}^m F_j(a_{i_q}) = 1, a_{i_q} \in A_k(F_j). \quad (2.72)$$

При КФС каждый отдельный элемент a_{i_q} тела $A_k(F_j)$ или $A_k(F_j)$ *влияет, но недостаточен* для реализации унитарного цвета $P \wedge S$ -множества – этот цвет реализуется только при наличии одновременно всех элементов тела данного цвета. Поэтому описание $P \wedge S$ -множества осуществляется на уровне (2.66) и содержит все компоненты (2.16):

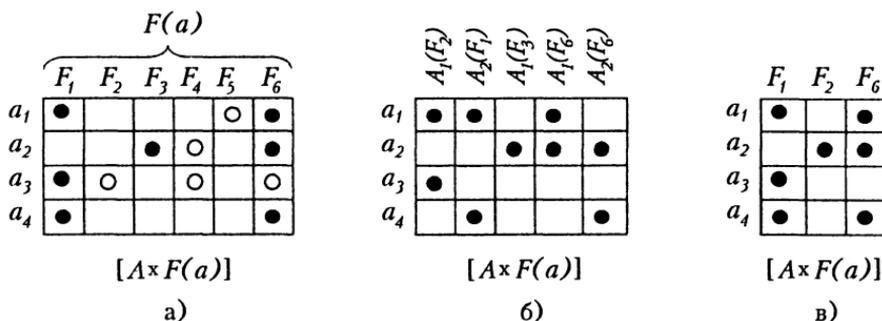
$$P \wedge S = (A, F(a), F(A), [A \times F(a)], [A \times F(A)], [A \times A(F)]). \quad (2.73)$$

Состав унитарных цветов в раскраске $F(A)$ $P \wedge S$ -множества определяется, исходя из условий (2.71), (2.72) существования таких цветов при наличии элементов, образующих тело данного цвета; варианты этих тел приводятся в булевой матрице $[A \times A(F)]$. Поскольку любой унитарный цвет $F_j(A)$ существует только при наличии хотя бы одного тела $A_k(F_j)$, то возможно, что некоторые цвета элементов $P \wedge S$ -множества не имеют одноименных унитарных цветов в составе $F(A)$ или не участвуют в их реализации. В этом случае соотношение (2.68) не выполняется и имеет место $F(A) \subset F(a)$. Пример такого $P \wedge S$ -множества приведен на (рис.2.2).

Если в составе персональных цветов элементов $P \wedge S$ -множества содержатся только цвета, непосредственно влияющие на существование одноименных унитарных цветов согласно условию (2.44), то описание $P \wedge S$ -множества будет достаточно полным на уровне (2.64), поскольку нет необходимости включать в описание множество $F(a)$ и матрицу (2.17), совпадающую с матрицей (2.10). В результате $P \wedge S$ -множество будет иметь минимальный набор компонентов:

$$P \wedge S = (A, F(A), [A \times F(A)], [A \times A(F)]). \quad (2.74)$$

Если PS -множество моделирует техническую или другую систему с определенными функциями, то в состав такого множества не

Рис.2.2. Пример PS -множества

а) матрица $[A \times F(a)]$ цветов элементов; б) матрица $[A \times A(F)]$ тел унитарных цветов; в) матрица $[A \times F(A)]$ унитарных цветов • – элементы $c_{j(j)} = 1$ булевых матриц; ○ – элементы $c_{i(j)} = 1$ матрицы $A \times F(a)$, соответствующие цветам элементов, не влияющих на унитарные цвета PS -множества

могут входить элементы с произвольной раскраской. Поскольку унитарная раскраска самого PS -множества зависит от персональных раскрасок входящих элементов, то условия существования любого элемента в данном множестве зависят от его раскраски и раскрасок других элементов этого PS -множества.

Возможность существования элемента a_k в конкретном PS -множестве в зависимости от наличия в этом множестве других элементов определяется взаимосвязью персональных цветов a_k и других элементов этого множества. С этой точки зрения все цвета $F(a_k)$ разделяются на две группы:

- совокупность собственных цветов $F^P(a_k)$, влияние которых на существование a_k в данном PS -множестве не зависит от персональных цветов других элементов;
- совокупность несобственных цветов $F^\Delta(a_k)$, влияющих на существование a_k в данном PS -множестве во взаимосвязи с персональными цветами других элементов этого множества.

Совокупность собственных цветов $F^P(a_k)$ разделяется на подмножество цветов

$$F_A^P(a_k) \subseteq F^P(a_k),$$

при наличии которых элемент a_k может входить в множество A элементов данного PS -множества и/или подмножество цветов

$$F_A^P(a_k) \subseteq F^P(a_k),$$

при наличии которых элемент a_k не может входить в это множество.

Отношение принадлежности элемента a_k к составу A элементов PS -множества отображается логической переменной

$$a_k = \begin{cases} 1, & \text{если } a_k \in A, \\ 0, & \text{если } a_k \notin A. \end{cases}$$

Взаимосвязь цветов $F^P(a_k)$ по условиям их существования представляется в булевой матрице (2.20). Условие, определяющее возможность существования a_k в данном PS -множестве в зависимости от состава цветов $F_A^P(a_k)$ и $F_{\bar{a}_k}^P(a_k)$, без учета других ограничений, будет таким:

$$a_k = 1, \text{ если } (F_A^P(a_k) \neq \emptyset) \wedge F_{\bar{a}_k}^P(a_k) = \emptyset = 1. \quad (2.75)$$

В раскраске $F(a_k)$ элемента a_k может существовать совокупность цветов, влияющих на существование a_k в PS -множестве во взаимосвязи с персональными цветами других элементов этого множества. Такие цвета элемента a_k и соответствующие им цвета других элементов, отображаемые в составе $F(a)$, будут *сопряженными*. Необходимые и достаточные условия существования элемента a_k в данном PS -множестве в зависимости от сопрягаемых цветов будут такими:

- среди сопряженных цветов в $F(a_k)$ и $F(a)$ должны существовать составы цветов $F_A^\Delta(a_k)$ и $F_{\bar{a}_k}^\Delta(a)$, взаимосвязь которых обеспечивает возможность существования a_k в данном PS -множестве;
- среди сопряженных цветов в $F(a_k)$ и $F(a)$ не должны существовать составы цветов $F_A^\Delta(a_k)$ и $F_{\bar{a}_k}^\Delta(a)$, взаимосвязь которых делает невозможным существование a_k в данном PS -множестве.

Рассмотрим сопряженные цвета $F_A^\Delta(a_k)$ и $F_{\bar{a}_k}^\Delta(a)$, обеспечивающие возможность существования элемента a_k в данном PS -множестве. Эти составы могут быть структурированными, состоящими из отдельных групп цветов:

$$F_A^\Delta(a_k)_i = (F_{i_1}, F_{i_2}, \dots, F_{i_{m_i}}) \subseteq F_A^\Delta(a_k),$$

$$F_{\bar{a}_k}^\Delta(a)_j = (F_{j_1}, F_{j_2}, \dots, F_{j_{m_j}}) \subseteq F_{\bar{a}_k}^\Delta(a),$$

таких, что каждая соответствующая пара сопряженных групп цветов обеспечивает возможность существования a_k в PS -множестве.

Составы цветов $F_A^\Delta(a_k)$ и $F_{\bar{a}_k}^\Delta(a)$ также могут быть структурированными, состоящими из отдельных групп цветов:

$$F_A^\Delta(a_k)_i = (F_{i_1}, F_{i_2}, \dots, F_{i_{m_i}}) \subseteq F_A^\Delta(a_k),$$

$$F_{\bar{a}_k}^\Delta(a)_j = (F_{j_1}, F_{j_2}, \dots, F_{j_{m_j}}) \subseteq F_{\bar{a}_k}^\Delta(a),$$

таких, что каждая соответствующая пара непосредственно сопряженных групп цветов делает невозможным существование a_k в данном PS -множестве.

Условие, определяющее возможность существования a_k за счет сопряжения его цветов с цветами других элементов данного PS -множества, будет таким

$$\left. \begin{aligned} a_k = 1, & \text{ если } \bigvee_{c=1}^n (F_A^\Delta(a_k)_i \wedge F_{a_k}^\Delta(a)_j)_c = 1 \\ a_k = 1, & \text{ если } \bigvee_{c=1}^n (F_A^\Delta(a_k)_i \wedge F_{a_k}^\Delta(a)_j)_c = 1 \end{aligned} \right\} \quad (2.76)$$

Условия, определяющие возможность существования элемента a_k в данном PS -множестве, могут быть описаны не только через составы сопряженных цветов (2.76), но и через составы элементов, обеспечивающих или препятствующих существованию a_k в этом PS -множестве.

2.1.3. Основные операции над полихроматическими множествами

Операции над PS -множествами, осуществляемые с учетом их унитарных раскрасок и персональных цветов входящих элементов, более сложны, чем операции над обычными множествами. Операции над PS -множествами не могут быть сведены к оперированию только составами элементов множеств, участвующих в операции, как это имеет место при выполнении операций над обычными множествами. Особенности операций над PS -множествами обусловлены необходимостью оперирования с составами не только их элементов, но и их унитарными цветами, связанных определенными отношениями с персональными цветами элементов этих PS -множеств.

Выполнение операций над PS -множествами должно осуществляться при условии, что и все исходные PS -множества, и итоговое PS -множество представлены на одинаковом уровне детализации их описания.

Для выполнения операций над PS -множествами описания всех их элементов должны быть приведены к нормализованному виду,

когда все элементы с различающимися свойствами имеют разные имена, а все элементы с одинаковыми свойствами – одноименны.

Изменения *PS*-множества связаны с изменением состава его элементов. Основными операциями являются:

- исключение (удаление) существовавшего в *PS*-множестве элемента;
- включение (добавление) в *PS*-множество нового элемента;
- замена в *PS*-множестве существовавшего ранее новым элементом. Замена является комбинацией последовательно выполняемых операций исключения существовавшего ранее и включения нового элемента *PS*-множества.

Изменение состава элементов *PS*-множества существенно зависит от допустимых условий его существования. Если на существование рассматриваемого *PS*-множества не накладывается никаких ограничений, то оно называется свободным. В состав свободного *PS*-множества может входить любой элемент.

Если существование *PS*-множества ограничено некоторыми условиями, то оно называется ограниченным. В состав такого *PS*-множества могут входить не любые, а лишь такие элементы, при наличии которых *PS*-множество удовлетворяет заданным условиям существования.

Если на первое место выдвигаются проблемы обеспечения требуемых свойств системы, то состав элементов системы рассматривается как средство получения требуемых свойств. Такие проблемы решаются с использованием операций над раскрасками *PS*-множества, к которым относятся:

- исключение (удаление) унитарного цвета, существовавшего в раскраске *PS*-множества;
- включение (добавление) нового унитарного цвета в раскраску *PS*-множества;
- замена существовавшего ранее в раскраске *PS*-множества унитарного цвета новым цветом. Замена является комбинацией последовательно выполняемых операций исключения существовавшего ранее и добавления нового унитарного цвета в раскраску *PS*-множества.

Операции над раскрасками *PS*-множеств осуществляются путем изменения состава его элементов.

Исключение элемента из свободного *PS*-множества является одной из наиболее простых операций, поскольку данные о свойствах исходного *PS*-множества достаточны для определения всех свойств получаемого *PS*-множества. Если исходное *PS*-множество имеет состав компонентов (2.16), то вначале определяется состав A_{-k}

исключением из A элемента a_k , а затем из булевых матриц (2.7), (2.10) и (2.14) исключаются строки, соответствующие элементу a_k . Далее определяются составы цветов $F(a)_{-k}$, $F(A)_{-k}$ и тел $A(F)_{-k}$, реализуемых за счет остающихся в A_{-k} элементов, и окончательно формируются новые булевы матрицы (2.7), (2.10) и (2.14). В результате получается новое PS -множество

$$PS_{-k} = (A_{-k}, F(a)_{-k}, F(A)_{-k}, [A_{-k} \times F(a)_{-k}], [A_{-k} \times F(A)_{-k}], [A_{-k} \times A(F)_{-k}]). \quad (2.77)$$

В множестве PS_{-k} могут исчезнуть некоторые цвета и тела унитарных цветов, существование которых в исходном PS -множестве было обусловлено существованием исключаемого элемента a_k . Элементы и цвета множества PS_{-k} связаны с цветами и элементами исходного PS -множества очевидными соотношениями:

$$A_{-k} \subset A; F(a)_{-k} \subseteq F(a); F(A)_{-k} \subseteq F(A); A(F)_{-k} \subset A(F). \quad (2.78)$$

Формируемое множество PS_{-k} может быть свободным или ограниченным по допускаемому составу элементов или унитарных цветов. В последнем случае в процесс формирования PS_{-k} включаются операции по проверке компонентов (2.77) на соответствие заданным условиям.

Цвета разных элементов PS -множества при представлении их в составе $F(a)$ и булевой матрице $[A \times F(a)]$ рассматриваются как независимые от цветов других элементов. Поэтому способы вычисления компонентов $F(a)_{-k}$ и $[A_{-k} \times F(a)_{-k}]$ множества (2.77) одинаковы для дизъюнктивных и конъюнктивных PS -множеств. В исходном PS -множестве состав цветов $F(a)$ вычисляется по формуле (2.8). После исключения элемента a_k состав цветов $F(a)_{-k}$ в новом PS_{-k} -множестве вычисляется по формуле

$$F(a)_{-k} = \bigvee_i^{n-1} F(a_i), \quad i \neq k; \quad (2.79)$$

при этом, очевидно,

$$F(a) = F(a)_{-k} \vee F(a_k). \quad (2.80)$$

Состав цветов $F(a)_{-k}$ разделяется на характерные группы. Две группы цветов определенным образом взаимосвязаны с цветами $F(a_k)$ исключаемого элемента. Группа цветов $F(a)_{-k}^I$ определяется как конъюнкция булева вектора $F(a_k)$ и инверсии булева вектора $F(a)_{-k}$:

$$F(a)_{-k}^I = \bar{F}(a)_{-k} \wedge F(a_k). \quad (2.81)$$

В эту группу входят те цвета, которые существуют в раскраске $F(a)$ исходного множества PS только за счет элемента a_k , и после удаления a_k в раскраске $F(a)_{-k}$ не существуют.

Состав цветов второй группы существует и в $F(a)$, и в $F(a)_{-k}$; этот состав определяется как конъюнкция булевых векторов $F(a)_{-k}$ и $F(a_k)$:

$$F(a)_{-k}^{II} = F(a)_{-k} \wedge F(a_k). \quad (2.82)$$

Цвета, не взаимосвязанные с раскраской $F(a_k)$ исключаемого элемента a_k , также разделяются на две характерные группы:

- цветов $F(a)_{-k}^{III}$, существующих в раскрасках $F(a)$ и $F(a)_{-k}$ только за счет других элементов исходного PS -множества;
- цветов $F(a)_{-k}^{IV}$, не существующих в раскрасках $F(a)$ и $F(a)_{-k}$ исходного PS -множества и конечного множества PS_{-k} .

Состав цветов группы $F(a)_{-k}^{III}$ определяется как разность множеств $F(a)_{-k}$ и $F(a_k)$:

$$F(a)_{-k}^{III} = F(a)_{-k} \setminus F(a_k), \quad (2.83)$$

или как конъюнкция булевых векторов $F(a)_{-k}$ и $\bar{F}(a_k)$:

$$F(a)_{-k}^{III} = F(a)_{-k} \wedge \bar{F}(a_k). \quad (2.84)$$

Всем цветам, входящим в группы $F(a)_{-k}^I$, $F(a)_{-k}^{II}$, $F(a)_{-k}^{III}$, соответствуют одноименные цвета в раскраске $F(a)$ исходного PS -множества, поэтому

$$F(a) = F(a)_{-k}^I \vee F(a)_{-k}^{II} \vee F(a)_{-k}^{III}. \quad (2.85)$$

Очевидно, состав всех цветов в раскраске конечного PS_{-k} -множества, вычисляемый по формуле (2.79), равен

$$F(a)_{-k} = F(a)_{-k}^{II} \vee F(a)_{-k}^{III} \quad (2.86)$$

Группа цветов $F(a)_{-k}^{IV}$, не существующих в раскраске $F(a)$ исходного PS -множества и в раскраске $F(a)_{-k}$ конечного множества PS_{-k} , вычисляется по формуле

$$F(a)_{-k}^{IV} = \bar{F}(a) = \bar{F}(a)_{-k} \wedge \bar{F}(a_k). \quad (2.87)$$

После вычисления состава цветов $F(a)_{-k}$ формируется булева матрица $[A_{-k} \times F(a)_{-k}]$ исключением из матрицы $[A \times F(a)]$

исходного PS -множества строки, соответствующей исключаемому элементу a_k , и столбцов, соответствующих цветам группы $F(a)_{-k}^I$, если $F(a)_{-k}^I \neq \emptyset$.

В отличие от компонентов $F(a)_{-k}$ и $[A_{-k} \times F(a)_{-k}]$, способы определения компонентов $F(A)_{-k}$, $[A_{-k} \times F(A)_{-k}]$ и $[A_{-k} \times A(F)_{-k}]$ множества (2.77) зависят от типа исходного PS -множества и логических отношений между унитарными цветами $F(A)$. Если исходным будет свободное дизъюнктивное PS -множество, то, согласно (2.68), имеет место равенство $F(A) = F(a)$. Если при этом унитарные цвета в $F(A)$ взаимно независимы – связаны дизъюнкцией, то составы $F(A)_{-k}$ и $F(a)_{-k}$ также будут равны:

$$F(A)_{-k} = F(a)_{-k}, \quad (2.88)$$

при этом, очевидно,

$$[A_{-k} \times F(A)_{-k}] = [A_{-k} \times F(a)_{-k}]. \quad (2.89)$$

Тем самым построение конечного множества PS_{-k} завершено, поскольку в этом случае нет необходимости формировать матрицу (2.14) вариантов тел унитарных цветов – роль этой матрицы выполняет матрица $[A_{-k} \times F(A)_{-k}]$.

В рассматриваемых $\Pi^{\vee}S$ -множествах между унитарными раскрасками $F(A)$ и $F(A)_{-k}$ исходного и конечного множеств, и раскраской $F(a_k)$ исключаемого элемента a_k существуют отношения, аналогичные отношениям (2.81)–(2.87) между раскрасками $F(a)$, $F(a)_{-k}$ и $F(a_k)$. Унитарные цвета, не реализуемые в раскраске $F(A)_{-k}$ множества $\Pi^{\vee}S_{-k}$, образуют группу цветов $F(A)_{-k}^I$, существовавших в раскраске $F(A)$ исходного множества только за счет элемента a_k , и исчезающих в $F(A)_{-k}$ при исключении a_k :

$$F(A)_{-k}^I = \overline{F(A)}_{-k} \wedge F(a_k) = \overline{F(a)}_{-k} \wedge F(a_k) = F(a)_{-k}^I. \quad (2.90)$$

Группа цветов, существующая в $F(A)$ и за счет a_k , и за счет других элементов $\Pi^{\vee}S$ -множества, вычисляется по формуле:

$$F(A)_{-k}^{II} = F(A)_{-k} \wedge F(a_k) = F(a)_{-k} \wedge F(a_k) = F(a)_{-k}^{II}. \quad (2.91)$$

Унитарные цвета, существующие в $F(A)$ и $F(A)_{-k}$ только за счет других элементов, не считая исключаемый элемент a_k , вычисляются по формуле:

$$F(A)_{-k}^{III} = F(A)_{-k} \wedge \overline{F(a_k)} = F(a)_{-k} \wedge \overline{F(a_k)} = F(a)_{-k}^{III}. \quad (2.92)$$

Следовательно, унитарная раскраска $F(A)$ исходного PS -множества, аналогично (2.85), состоит из совокупности цветов групп $F(A)_{-k}^I$, $F(A)_{-k}^{II}$ и $F(A)_{-k}^{III}$,

$$F(A) = F(A)_{-k}^I \vee F(A)_{-k}^{II} \vee F(A)_{-k}^{III},$$

а унитарная раскраска $F(A)_{-k}$ множества PS_{-k} – из цветов групп $F(A)_{-k}^{II} \vee F(A)_{-k}^{III}$

$$F(A)_{-k} = F(A)_{-k}^{II} \vee F(A)_{-k}^{III}. \quad (2.93)$$

Унитарные цвета, вообще не существовавшие в раскраске $F(A)$ исходного $P^{\vee}S$ -множества, а потому не существующие и в раскраске $P^{\vee}S_{-k}$, вычисляются по формуле:

$$F(A)_{-k}^{IV} = \bar{F}(A)_{-k} \wedge \bar{F}(a_k) = \bar{F}(a)_{-k} \wedge \bar{F}(a_k) = F(a)_{-k}^{IV} = \bar{F}(a). \quad (2.94)$$

Если в любом $P^{\vee}S$ -множестве, согласно (2.68), всем цветам $F(a_k)$ раскраски элемента a_k соответствуют одноименные унитарные цвета в раскраске $F(A)$, то в $P^{\wedge}S$ -множестве любому цвету $F_j(a_k)$ соответствует одноименный цвет $F_j(A)$ только тогда, когда существует хотя бы одно тело $A_k(F_j)$; если ни одного такого тела не существует, то и унитарного цвета, одноименного $F_j(a_k)$, не существует. Следовательно, в $P^{\wedge}S$ -множестве раскраска $F(a_k)$ любого элемента a_k может состоять из двух частей:

$$F(a_k) = F_A(a_k) \cup F_{\bar{A}}(a_k),$$

где $F_A(a_k)$ – подмножество цветов, которым в раскраске $F(A)$ соответствуют одноименные унитарные цвета; $F_{\bar{A}}(a_k)$ – подмножество цветов a_k , для которых в раскраске $F(A)$ не существуют одноименные унитарные цвета. Очевидно,

$$F_A(a_k) = F(A) \wedge F(a_k), F_A(a_k) \subseteq F(A), \quad (2.95)$$

$$F_{\bar{A}}(a_k) = \bar{F}(A) \wedge F(a_k), F_{\bar{A}}(a_k) \not\subseteq F(A). \quad (2.96)$$

Поскольку данные о непосредственном участии элемента a_k в реализации унитарных цветов $P^{\wedge}S$ -множества имеют существенное значение, состав цветов $F_A(a_k)$, одноименных унитарным цветам в $F(A)$, целесообразно разделять на две части – $F_A(a_k)^P$ и $F_A(a_k)^N$, обладающие следующими свойствами:

$$\forall F_i \in F_A(a_k)^P [\exists A_p(F_i), a_k \in A_p(F_i)], \quad (2.97)$$

$$\forall F_j \in F_A(a_k)^N [\forall A_q(F_j), a_k \notin A_q(F_j)]. \quad (2.98)$$

Цвета, обладающие свойством (2.97), входят в группу $F(a)_{-k}^I$ или $F(a)_{-k}^{II}$, а цвета, обладающие свойством (2.98) – в группу $F(a)_{-k}^{III}$.

Состав унитарной раскраски $F(A)_{-k}$ после исключения a_k из исходного $\Pi \wedge S$ -множества определяется так:

- из булевой матрицы $[A \times A(F)]$ вычеркивается строка, соответствующая исключаемому элементу a_k , и столбцы, соответствующие телам, в которые входит a_k .
- определяется состав $A(F)_{-k}$ столбцов, оставшихся после исключения элемента a_k , и формируется булева матрица тел $[A_{-k} \times A(F)_{-k}]$ нового множества $\Pi \wedge S_{-k}$.
- определяется состав $F(A)_{-k}$ унитарных цветов, существование которых обеспечено наличием соответствующих тел в матрице $[A_{-k} \times A(F)_{-k}]$.

После этого из булевой матрицы $[A \times F(A)]$ исходного $\Pi \wedge S$ -множества вычеркивается строка, соответствующая a_k , и столбцы, соответствующие унитарным цветам, для которых в матрице $[A_{-k} \times A(F)_{-k}]$ нет ни одного тела; оставшаяся часть исходной матрицы $[A \times F(A)]$, имеющая состав $F(A)_{-k}$ унитарных цветов и состав A_{-k} элементов, будет матрицей $[A_{-k} \times F(A)_{-k}]$. Тем самым формирование нового множества $\Pi \wedge S_{-k}$ завершено, поскольку составы компонентов $F(a)_{-k}$ и $[A_{-k} \times F(a)_{-k}]$ определены ранее. В полученном $\Pi \wedge S_{-k}$ -множестве имеет место соотношение

$$F(A)_{-k} \subseteq F(a)_{-k}. \quad (2.99)$$

В общем случае диаграммы Венна, характеризующие соотношения раскрасок $F(A)$, $F(a)$, $F(A)_{-k}$ и $F(a)_{-k}$ при исключении элемента a_k из исходного множества $\Pi \wedge S$, имеют вид (рис.2.3). На рис.2.3а показаны диаграммы раскраски исходного $\Pi \wedge S$ -множества, а на рис.2.3б – диаграммы раскраски $\Pi \wedge S_{-k}$ после исключения элемента a_k .

Цвета раскрасок исходного множества $\Pi \wedge S$ и получаемого множества $\Pi \wedge S_{-k}$ по отношению к раскраске исключаемого из исходного $\Pi \wedge S$ -множества элемента a_k разделяются на характерные группы, расположенные в разных зонах диаграмм Венна (рис.2.3). Группе

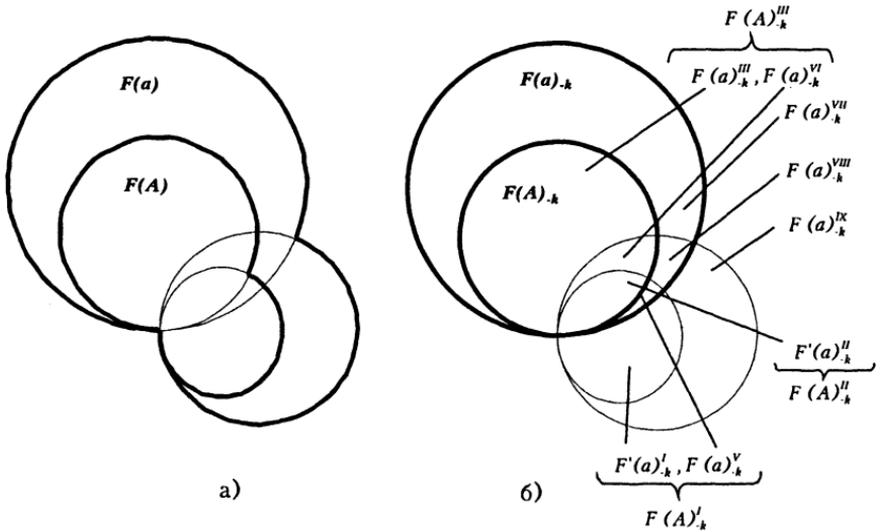


Рис.2.3. Диаграммы Венна при исключении элемента a_k из $\Pi \wedge S$ -множества

цветов $F(A)_{-k}^I$, существовавших в исходном $\Pi \wedge S$ -множестве только за счет элемента a_k , и исчезающих в $F(A)_{-k}$ при исключении a_k , соответствуют две зоны в диаграммах Венна:

$$F(A)_{-k}^I = \bar{F}(A)_{-k} \wedge F_A(a_k)^P = F'(a)_{-k}^I \wedge F(a)_{-k}^V, \tag{2.100}$$

где цвета

$$F'(a)_{-k}^I = \bar{F}(a)_{-k} \wedge F_A(a_k)^P \tag{2.101}$$

не имеют одноименных цветов в $F(a)_{-k}$, и цвета

$$F(a)_{-k}^V = F(a)_{-k} \wedge F_A(a_k)^P \wedge \bar{F}(A)_{-k}, \tag{2.102}$$

для которых в $F(a)_{-k}$ имеются одноименные цвета, по телом любого унитарного цвета, одноименного цветам $F(a)_{-k}^V$, как и $F'(a)_{-k}^I$, является только элемент a_k . Поэтому здесь состав цветов $F(A)_{-k}^I$ отличается от состава вычисляемого по формуле (2.90)

Группа цветов $F(A)_{-k}^{II}$, существовавших в исходном $\Pi \wedge S$ -множестве либо за счет a_k и других элементов A , либо только за счет a_k , здесь вычисляется не по формуле (2.91), а по формуле:

$$F(A)_{-k}^{II} = F'(a)_{-k}^{II} = F(A)_{-k} \wedge F_A(a_k)^P. \tag{2.103}$$

В составе цветов $F(a)_{-k}$, одноименных унитарным цветам $F(A)_{-k}$, могут быть цвета, одноименные цветам $F(a_k)$, не входящим в $F_A(a_k)^P$, вычисляемые по формуле:

$$F(a)_{-k}^{VI} = F(A)_{-k} \wedge F(a_k) \wedge \bar{F}_A(a_k)^P. \quad (2.104)$$

Состав цветов, вычисляемый по этой формуле, в раскраске $F(a_k)$ соответствует цветам $F_A(a_k)^N$. Поэтому группа цветов $F(A)_{-k}^{III}$, существующих в $F(A)$ и $F(A)_{-k}$ только за счет других элементов, не считая исключаемый элемент a_k , вычисляется по формуле:

$$F(A)_{-k}^{III} = F(a)_{-k}^{III} \vee F(a)_{-k}^{VI}, \quad (2.105)$$

где $F(a)_{-k}^{III}$ вычисляется по формуле:

$$F(a)_{-k}^{III} = F(A)_{-k} \wedge \bar{F}(a_k), \quad (2.106)$$

соответствующей формуле (2.92).

Группа цветов $F(A)_{-k}^{IV}$, вообще не существовавших в раскраске $F(A)$ исходного $\Pi \wedge S$ -множества, а поэтому не существующих и в раскраске $F(A)_{-k}$ множества $\Pi \wedge S_{-k}$, вычисляется по формуле:

$$F(A)_{-k}^{IV} = \bar{F}(a) \vee F_{\bar{A}}(a), \quad (2.107)$$

где состав $F_{\bar{A}}(a)$ определяется по формуле $F_{\bar{A}}(a) = \bar{F}(A) \wedge F(a)$. Поскольку, согласно (рис.2.36),

$$F_{\bar{A}}(a) = F(a)_{-k}^{VII} \vee F(a)_{-k}^{VIII} \vee F(a)_{-k}^{IX}, \quad (2.108)$$

то

$$F(A)_{-k}^{IV} = \bar{F}(a) \vee F(a)_{-k}^{VII} \vee F(a)_{-k}^{VIII} \vee F(a)_{-k}^{IX}, \quad (2.109)$$

где

$$F(a)_{-k}^{VII} = F_{\bar{A}}(a) \wedge \bar{F}(a_k) = F(a)_{-k} \wedge \bar{F}(a_k) \wedge \bar{F}(A)_{-k}, \quad (2.110)$$

$$F(A)_{-k}^{VIII} = F(a)_{-k} \wedge F(a_k) \wedge \bar{F}(A)_{-k} \wedge \bar{F}_A(a_k)^P, \quad (2.111)$$

$$F(a)_{-k}^{IX} = \bar{F}(a)_{-k} \wedge F(a_k) \wedge \bar{F}_A(a_k)^P. \quad (2.112)$$

В составе $\bar{F}(a)$ содержатся цвета из раскраски $F(U)$ универсума, не входящие в раскраски ни одного элемента исходного $\Pi \wedge S$ -множества. В состав $F(a)_{-k}^{VII}$, $F(a)_{-k}^{VIII}$ и $F(a)_{-k}^{IX}$ входят цвета элементов

исходного $\Pi^{\wedge}S$ -множества, не имеющие одноименных унитарных цветов в раскраске $F(A)$; при этом $F(a)_{-k}^{VII}$ содержит такие цвета других элементов, исключая цвета $F_A^-(a_k)$ элемента a_k , $F(a)_{-k}^{VIII}$ содержит цвета, имеющиеся и в $F_A^-(a_k)$, и в раскрасках других элементов, а $F(a)_{-k}^{IX}$ содержит только цвета из раскраски $F_A^-(a_k)$.

При добавлении в свободное ΠS -множество с составом компонентов (2.16) элемента a_k , имеющего раскраску $F(a_k)$, получается новое множество:

$$\begin{aligned} \Pi S_{+k} = & (A_{+k}, F(a)_{+k}, F(A)_{+k}, [A_{+k} \times F(a)_{+k}], \\ & [A_{+k} \times F(A)_{+k}], [A_{+k} \times A(F)_{+k}]). \end{aligned} \quad (2.113)$$

В отличие от операции исключения элемента из ΠS -множества, при добавлении в это множество нового элемента a_k в получаемом множестве ΠS_{+k} сохраняются составы всех элементов и цветов, существовавших в исходном ΠS -множестве, и могут появиться новые цвета, обусловленные появлением нового элемента a_k . Новые цвета могут появиться не только в $F(a)_{+k}$, но и в унитарной раскраске $F(A)_{+k}$ из-за того, что появление a_k может привести к появлению новых тел унитарных цветов. Поэтому элементы и цвета ΠS_{+k} связаны с элементами и цветами исходного ΠS -множества соотношениями:

$$A_{+k} \supset A; F(a)_{+k} \supseteq F(a); F(A)_{+k} \supseteq F(A); A(F)_{+k} \supseteq A(F). \quad (2.114)$$

Поскольку цвета разных элементов ΠS -множества при представлении их в составе $F(a)$ и булевой матрице $[A \times F(a)]$ рассматриваются как независимые от цветов других элементов, способы вычисления компонентов $F(a)_{+k}$ и $[A_{+k} \times F(a)_{+k}]$ нового множества (2.113) одинаковы для дизъюнктивных и конъюнктивных множеств.

В исходном ΠS -множестве состав цветов $F(a)$ вычисляется по формуле (2.6). После добавления элемента a_k состав цветов $F(a)_{+k}$ в новом множестве ΠS_{+k} вычисляется по формуле:

$$F(a)_{+k} = F(a) \cup F(a_k), \quad (2.115)$$

или, при представлении всех цветов в едином булевом векторном пространстве (2.5), по формуле:

$$F(a)_{+k} = F(a) \vee F(a_k). \quad (2.116)$$

Для построения булевой матрицы $[A_{+k} \times F(a)_{+k}]$ в матрицу $[A \times F(a)]$ исходного ΠS -множества добавляется строка a_k и столбцы, соответствующие новым цветам, появившимся в $F(a)_{+k}$. Затем отмечаются элементы $c_{\bar{k}(j)} = 1$, соответствующие цветам $F_j \in F(a_k)$, и построение матрицы $[A_{+k} \times F(a)_{+k}]$ нового множества завершено.

Цвета раскраски $F(a)_{+k}$, получаемой после включения элемента a_k в исходное PS -множество, разделяются на группы:

– цветов $F(a)_{-k}^I$, существующих в раскраске $F(a)_{+k}$ только за счет нового элемента a_k :

$$F(a)_{+k}^I = \bar{F}(a) \wedge F(a_k); \quad (2.117)$$

– цветов $F(a)_{-k}^{II}$, существующих в $F(a)_{+k}$ и за счет a_k , и за счет других элементов исходного PS -множества:

$$F(a)_{-k}^{II} = F(a) \wedge F(a_k); \quad (2.118)$$

– цветов $F(a)_{-k}^{III}$, существующих в $F(a)_{+k}$ только за счет элементов исходного PS -множества:

$$F(a)_{-k}^{III} = F(a) \wedge \bar{F}(a_k). \quad (2.119)$$

Общий состав цветов $F(a)_{+k}$ множества PS_{+k} является объединением всех цветов, входящих в группы (2.117)–(2.119):

$$F(a)_{+k} = F(a)_{-k}^I \vee F(a)_{+k}^{II} \vee F(a)_{-k}^{III}. \quad (2.120)$$

При решении некоторых задач представляет интерес и группа цветов $F(a)_{-k}^{IV}$, не существующих ни в $F(a)$, ни в $F(a)_{+k}$:

$$F(a)_{+k}^{IV} = \bar{F}(a) \wedge \bar{F}(a_k) = \bar{F}(a)_{+k}. \quad (2.121)$$

После вычисления состава цветов $F(a)_{+k}$ по формуле (2.115) или (2.116) формируется булева матрица $[A_{+k} \times F(a)_{+k}]$ путем добавления в матрицу $[A \times F(a)]$ исходного PS -множества строки, соответствующей добавляемому элементу a_k , и столбцов, соответствующих цветам, входящим в группу $F(a)_{+k}^I$.

Способы определения компонентов $F(A)_{+k} [A_{+k} \times F(A)_{+k}]$ и $[A_{+k} \times A(F)_{+k}]$ получаемого множества (2.113) зависят от типа исходного PS -множества и логических отношений между унитарными цветами получаемого множества PS_{+k} , а не исходного PS -множества, как при исключении элемента a_k . При исключении a_k характер исходного логического уравнения (2.53) не изменяется – меняются только истинностные значения цветов, входящих в группу (2.81), и зависящих от них унитарных цветов исходного PS -множества. При включении в PS -множество нового элемента a_k , при $F(a)_{+k}^I \neq 0$, в уравнении (2.53) для унитарных цветов $F(A)_{+k}$ получаемого

множества PS_{+k} могут появиться новые цвета, не существовавшие в составе $F(A)$ унитарных цветов исходного PS -множества. Логические отношения между новыми и существовавшими в $F(A)$ цветами неизвестны, поэтому такие отношения должны быть заданы заранее или определены в процессе формирования PS_{+k} .

Если исходным будет свободное дизъюнктивное $\Pi^{\vee}S$ -множество, то имеет место равенство $F(A) = F(a)$. Если при этом унитарные цвета (2.53) взаимно независимы и в $F(A)$, и в любом другом составе унитарных цветов, включая $F(A)_{+k}$, то составы $F(A)_{+k}$ и $F(a)_{+k}$ также будут равны:

$$F(A)_{+k} = F(a)_{+k},$$

причем

$$F(A)_{+k} = F(A) \vee F(a_k). \quad (2.122)$$

При этом будет иметь место и равенство булевых матриц:

$$[A_{+k} \times F(A)_{+k}] = [A_{+k} \times F(a)_{+k}]. \quad (2.123)$$

Тем самым, построение конечного множества $\Pi^{\vee}S_{+k}$ завершено, поскольку в этом случае нет необходимости формировать матрицу (2.14) тел унитарных цветов – роль этой матрицы выполняет матрица $[A_{+k} \times F(A)_{+k}]$.

В рассматриваемых $\Pi^{\vee}S$ -множествах между унитарными раскрасками $F(A)$ и $F(A)_{+k}$ исходного и конечного множеств и раскраской $F(a_k)$ включаемого в $\Pi^{\vee}S_{+k}$ элемента a_k существуют отношения, аналогичные отношениям (2.117)–(2.121) между раскрасками $F(a)$, $F(a)_{+k}$ и $F(a_k)$. Унитарные цвета $F(A)_{+k}$ разделяются на следующие группы:

– цветов $F(A)_{-k}^I$, существующих в $F(A)_{+k}$ только за счет нового элемента a_k :

$$F(A)_{-k}^I = \bar{F}(A) \wedge F(a_k); \quad (2.124)$$

– цветов $F(A)_{-k}^{II}$, существующих в $F(A)_{+k}$ за счет a_k , и других элементов исходного $\Pi^{\vee}S$ -множества:

$$F(A)_{-k}^{II} = F(A) \wedge F(a_k); \quad (2.125)$$

– цветов $F(A)_{-k}^{III}$, существующих в $F(A)_{+k}$ только за счет элементов исходного $\Pi^{\vee}S$ -множества:

$$F(A)_{-k}^{III} = F(A) \wedge \bar{F}(a_k). \quad (2.126)$$

Группа цветов $F(A)_{-k}^{IV}$, не существующих, ни в $F(A)$, ни в $F(A)_{+k}$, вычисляется по формуле

$$F(A)_{+k}^{IV} = \bar{F}(A) \wedge \bar{F}(a_k). \quad (2.127)$$

Общий состав унитарных цветов $F(A)_{+k}$ является объединением всех цветов, входящих в группы (2.124)–(2.126):

$$F(A)_{+k} = F(A)_{+k}^I \vee F(A)_{+k}^{II} \vee F(A)_{+k}^{III}. \quad (2.128)$$

При включении нового элемента a_k в конъюнктивное $\Pi^{\wedge}S$ -множество компоненты A_{+k} , $F(a)_{+k}$ и $[A_{+k} \times F(a)_{+k}]$ конечного множества $\Pi^{\wedge}S_{+k}$ вычисляются так же, как в дизъюнктивных $\Pi^{\vee}S$ -множествах. Способы вычисления компонентов $F(A)_{+k}$, $[A_{+k} \times F(A)_{+k}]$ и $[A_{+k} \times A(F)_{+k}]$ будут иными, и зависят от характера взаимосвязи унитарных цветов в $F(A)$ и состава элементов, входящих в тела унитарных цветов.

Поскольку включение в $\Pi^{\wedge}S$ -множество нового элемента a_k может привести и к появлению новых вариантов тел унитарных цветов, существующих в $F(A)$, и к появлению тел, реализующих новые унитарные цвета, до этого не существовавшие в $F(A)$, то возникает проблема *распознавания* неизвестных ранее тел унитарных цветов, появившихся в связи с наличием элемента $a_k \in A_{+k}$. Одним из путей решения этой проблемы является использование базы данных о полихроматическом универсуме ΠU , в котором должны быть представлены все известные тела всех известных унитарных цветов, реализуемых элементами данной предметной области. Реальным прообразом такого универсума являются архивы патентных служб. Другой путь, органически связанный с первым, – использование методов поиска новых технических решений.

Для вычисления нового множества $\Pi^{\wedge}S_{+k}$ к составу компонентов (2.16) исходного $\Pi^{\wedge}S$ -множества добавляется булева матрица:

$$[A^* \times A(F)]_U, A^* \supset A, a_k \in A^*. \quad (2.129)$$

Эта матрица формируется на основе базы данных об универсуме ΠU для рассматриваемой предметной области. Компоненты $F(A)_{+k}$, $[A_{+k} \times F(A)_{+k}]$ и $[A_{+k} \times A(F)_{+k}]$ вычисляются по такой схеме:

– в булеву матрицу $[A \times A(F)]$ исходного $\Pi^{\wedge}S$ -множества добавляется строка a_k и столбцы новых тел из матрицы (2.129), в которые входит элемент a_k и другие элементы из A ; в результате будет построена булева матрица $[A_{+k} \times A(F)_{+k}]$ тел унитарных цветов формируемого множества $\Pi^{\wedge}S_{+k}$;

- определяется состав унитарных цветов, реализуемых телами, входящими в матрицу $[A_{+k} \times F(A)_{+k}]$; этот состав и будет раскраской $F(A)_{+k}$ множества $\Pi^{\wedge}S_{+k}$.
- в булеву матрицу $[A \times F(A)]$ исходного $\Pi^{\wedge}S$ -множества добавляется строка a_k и столбцы, соответствующие новым унитарным цветам, появившимся в $F(A)_{+k}$; в результате будет построена матрица $[A_{+k} \times F(A)_{+k}]$ унитарных цветов нового множества $\Pi^{\wedge}S_{+k}$.

В результате множество $\Pi^{\wedge}S_{+k}$ будет полностью определено, поскольку компоненты A_{+k} , $F(a)_{+k}$ и $[A_{+k} \times F(a)_{+k}]$ вычисляются ранее.

Раскраска включаемого в $\Pi^{\wedge}S$ элемента a_k также разделяется на части $F_A(a_k)$ и $F_A^-(a_k)$, вычисляемые по формулам (2.95), (2.96). В свою очередь, состав цветов $F_A(a_k)$ разделяется на группу цветов $F_A(a_k)^P$, обладающих свойством (2.97), и группу цветов $F_A(a_k)^N$, обладающих свойством (2.98). Хотя цвета обеих групп соответствуют одноименным унитарным цветам конечного множества $\Pi^{\wedge}S_{+k}$, цвета группы $F_A(a_k)^P$ влияют на существование одноименных унитарных цветов $F(A)_{+k}$ и a_k входит в составы тел этих цветов, а цвета группы $F_A(a_k)^N$ не влияют на существование одноименных унитарных цветов $F(A)_{+k}$ и a_k не входит в составы тел этих цветов.

Цвета исходного множества $\Pi^{\wedge}S$, конечного множества $\Pi^?S_{+k}$ и включаемого в $\Pi^?S$ элемента a_k также разделяются на характерные группы, расположенные в разных зонах диаграмм Венна (рис.2.4).

Группе цветов $F(A)_{+k}^I$, существующей в $F(A)_{+k}$ только за счет нового элемента a_k , соответствуют две зоны в диаграмма Венна: зона $F(A)_{+k}^I$, вычисляемая по формуле

$$F(A)_{+k}^I = \bar{F}(A) \wedge F_A(a_k)^P = F'(a)_{+k}^I \vee F(a)_{+k}^V, \quad (2.130)$$

где

$$F'(a)_{+k}^I = \bar{F}(a) \wedge F_A(a_k)^P, \quad (2.131)$$

и зона $F(a)_{+k}^V$, вычисляемая по формуле

$$F(a)_{+k}^V = F(a) \wedge F_A(a_k)^P \wedge \bar{F}(A). \quad (2.132)$$

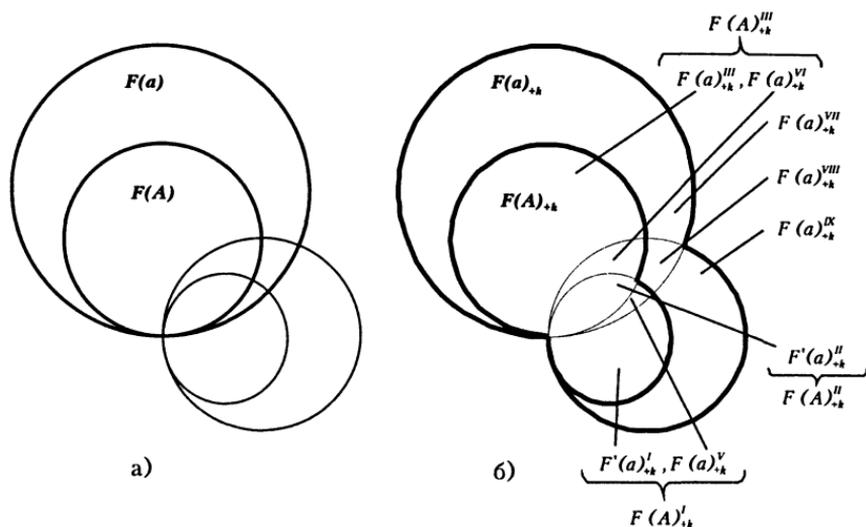


Рис.2.4. Соотношение раскрасок $F(A)$, $F(A)_{+k}$ и $F(a)_{+k}$ при включении элемента a_k в $\Pi \wedge S$ -множество при условии $F(A)F_1 \vee F_2 \vee \dots \vee F_m$

Сравнение этих формул с формулами (2.100), (2.101) и (2.102) показывает содержательное различие между ними: при исключении элемента a_k зоны $F'(a)_{-k}^I$ и $F(a)_{-k}^V$ определяются в зависимости от раскрасок $F(a)_{-k}$ и $F(A)_{-k}$ конечного множества $\Pi \wedge S_{-k}$, а при включении элемента a_k – в зависимости от раскрасок $F(a)$ и $F(A)$ исходного множества $\Pi \wedge S$. Содержание же самих зон в обоих случаях аналогично: тела унитарных цветов, одноименных цветам из $F'(a)_{+k}^I$ и $F(a)_{+k}^V$, состоят из одного элемента a_k ; при этом в составе $F(a)$ отсутствуют цвета других элементов, одноименные цветам $F'(a)_{+k}^I$, но есть цвета, одноименные цветам $F(a)_{+k}^V$.

Группа цветов $F(A)_{+k}^{II}$, существующих в $F(A)_{+k}$ за счет элемента a_k и других элементов исходного $\Pi \wedge S$ -множества, вычисляется по формуле:

$$F(A)_{+k}^{II} = F(A) \wedge F_A(a_k)^P = F'(a)_{+k}^{II}. \quad (2.133)$$

В составе цветов $F(a)_{+k}$, одноименных унитарным цветам $F(A)_{+k}$, могут быть цвета, одноименные цветам $F(a_k)$, не входящим в $F_A(a_k)^P$, вычисляемые по формуле:

$$F(a)_{+k}^{VI} = F(A) \wedge F(a_k) \wedge \bar{F}_A(a_k)^P. \quad (2.134)$$

Группа цветов $F(A)_{+k}^{III}$, существующих в $F(A)$ и $F(A)_{+k}$ только за счет других элементов, не считая присоединяемый элемент a_k , вычисляется по формуле:

$$F(A)_{+k}^{III} = F(a)_k^{III} \vee F(a)_{+k}^{VI}, \quad (2.135)$$

где $F(a)_{+k}^{III}$ вычисляется по формуле

$$F(a)_{+k}^{III} = F(A) \wedge \bar{F}(a_k) = F(A)_{+k} \wedge \bar{F}'(a)_{+k}^{II} \wedge \bar{F}(a)_{+k}^{VI}. \quad (2.136)$$

Общий состав унитарных цветов конечного множества $\Pi^2 S_{+k}$ является объединением всех цветов, входящих в группы $F(A)_{+k}^I$, $F(A)_{+k}^{II}$ и $F(A)_{+k}^{III}$, и вычисляется по формуле (3.120).

Группа цветов $F(A)_{+k}^{IV}$, вообще не существующих ни в $F(A)$, ни в $F(A)_{+k}$, состоит из цветов $\bar{F}(a)_{+k}$, не имеющих одноименных унитарных цветов в раскраске $\bar{F}(A)_{+k}$:

$$F(A)_{+k}^{IV} = \bar{F}(a)_{+k} \vee \bar{F}(A)_{+k} \quad (2.137)$$

или

$$F(A)_{+k}^{IV} = \bar{F}(a)_{+k} \vee F(a)_{+k}^{VII} \vee F(a)_{+k}^{VIII} \vee F(a)_{+k}^{IX}. \quad (2.138)$$

В этой формуле:

$$F(a)_{+k}^{VII} = F(a) \wedge \bar{F}(a_k) \wedge \bar{F}(A), \quad (2.139)$$

$$F(a)_{+k}^{VIII} = F(a) \wedge F(a_k) \wedge \bar{F}(A) \wedge \bar{F}_A(a_k)^P, \quad (2.140)$$

$$F(a)_{+k}^{IX} = \bar{F}(a_k) \wedge F(a_k) \wedge \bar{F}_A(a_k)^P. \quad (2.141)$$

Рассмотрим PS -множество с составом компонентов (2.16) и множество PS_i с составом компонентов

$$PS_i = (A_i, F(a)_i, F(A)_i, [A_i \times F(a)_i], [A_i \times F(A)_i], [A_i \times A(F)_i]), \quad (2.142)$$

которое будет подмножеством PS и обозначается $PS_i \subseteq PS$. Состав компонентов (2.142) подмножества PS_i определяется как результат исключения из PS -множества (2.16) элементов $a_k \in A$, не вошедших в состав A_i и образующих подмножество

$$A_i^{\complement} = A \setminus A_i. \quad (2.143)$$

Подмножество PS_i выделяется из исходного PS -множества по такой схеме:

- определяется состав элементов A_i , удовлетворяющий заданным условиям.
- вычисляется состав A_i^{\complement} элементов исходного PS -множества, не входящих в выделяемое подмножество PS_i . Если составы A и A_i представляются как булевы векторы, то

$$A_i^{\complement} = A \wedge \bar{A}_i. \quad (2.144)$$

- из исходного PS -множества методами, изложенными выше, исключаются элементы A_i^{\complement} . Оставшаяся после исключения A_i^{\complement} часть исходного PS -множества и будет выделяемым подмножеством PS_i .

В отличие от операции выделения подмножества, при вычислении пересечения, разности и объединения одновременно участвуют разные PS -множества, поэтому возникает проблема идентификации элементов этих множеств с целью выявления эквивалентных элементов. Оказавшись в одном и том же PS -множестве, все эквивалентные элементы – в силу закона идемпотентности – будут поглощены и представлены единственным элементом, который будет представлять собой класс эквивалентных элементов.

Для вычисления пересечения, разности и объединения требуется, чтобы и описания исходных PS -множеств, и описания их элементов были представлены на одинаковых уровнях детализации.

Пересечение обычных множеств A_i, A_j вычисляется по формуле:

$$A_{ij}^{\cap} = A_i \cap A_j. \quad (2.145)$$

Пересечение полихроматических множеств PS_i и PS_j формируется с учетом и состава элементов, и состава цветов этих множеств. Пересечение

$$PS_{ij}^{\cap} = PS_i \cap PS_j \quad (2.146)$$

можно рассматривать как подмножество, которое содержится одновременно в обоих исходных множествах PS_i и PS_j . Состав компонентов PS_{ij}^{\cap} :

$$PS_{ij}^{\cap} = (A_{ij}^{\cap}, F(a)_{ij}^{\cap}, F(A)_{ij}^{\cap}, [A_{ij}^{\cap} \times F(a)]),$$

$$[A_{ij}^{\cap} \times F(A)_{ij}^{\cap}], [A_{ij}^{\cap} \times A(F)_{ij}^{\cap}] \quad (2.147)$$

тесно связан со структурой обоих исходных множеств PS_i и PS_j .

Состав унитарных цветов $F(A)_{ij}$ пересечения соотносится с составами унитарных цветов исходных PS -множеств как подмножество во их пересечения:

$$F(A)_{ij}^{\cap} \subseteq F(A)_i \cap F(A)_j. \quad (2.148)$$

Унитарную раскраску $F(A)_{ij}^{\cap}$ можно определить, исходя из вычисленного по формуле (2.145) состава элементов A_{ij}^{\cap} , раскраски которых описаны в исходных множествах PS_i и PS_j . Если исходные множества оба дизъюнктивные, то их пересечение будет так же дизъюнктивным множеством $P^{\vee}S_{ij}^{\cap}$. Пересечение $P^{\vee}S_{ij}^{\cap}$ можно рассматривать как подмножество исходного множества $P^{\vee}S_i$ или $P^{\vee}S_j$.

Если исходные PS -множества – конъюнктивные, то пересечение этих множеств будет конъюнктивным множеством $P^{\wedge}S_{ij}^{\cap}$, и вычисляется так:

- составы компонентов исходных множеств $P^{\wedge}S_i$ и $P^{\wedge}S_j$ приводятся к единому уровню детализации их описания;
- описания всех элементов $P^{\wedge}S_i$ и $P^{\wedge}S_j$ приводятся к нормализованному виду;
- сравнивая попарно элементы исходных множеств, определяют эквивалентные элементы, входящие одновременно в $P^{\wedge}S_i$ и $P^{\wedge}S_j$, – эти элементы и образуют состав элементов A_{ij}^{\cap} пересечения $P^{\wedge}S_{ij}^{\cap}$;
- по формуле (2.6) или (2.8) определяют состав $F(a)_{ij}^{\cap}$ всех цветов элементов A_{ij}^{\cap} ;
- в матрицах $[A_i \times A(F)_i]$ и $[A_j \times A(F)_j]$ исходных множеств выявляются тела унитарных цветов, состоящие из элементов пересечения $P^{\wedge}S_{ij}^{\cap}$. Состав этих тел определяет унитарную раскраску $F(A)_{ij}$ пересечения $P^{\wedge}S_{ij}^{\cap}$;
- формируются булевы матрицы $P^{\wedge}S_{ij}^{\cap}$:

$$[A_{ij}^{\cap} \times F(a)_{ij}^{\cap}], [A_{ij}^{\cap} \times F(A)_{ij}^{\cap}], [A_{ij}^{\cap} \times A(F)_{ij}^{\cap}].$$

Рассмотрим, например, множества $\Pi^{\wedge}S_i$ и $\Pi^{\wedge}S_j$, составы булевых матриц (2.7), (2.10) и (2.14) которых приведены на рис.2.5а. Если эти множества рассматривать на уровне детализации описания (2.66), то из матриц $[A_i \times F(A)_i]$ и $[A_j \times F(A)_j]$ следует, что в составе элементов $\Pi^{\wedge}S_i$ и $\Pi^{\wedge}S_j$ нет эквивалентных элементов. Даже элементы $a_3 \in A_i$ и $a_{10} \in A_j$ не эквивалентны: хотя из матриц $[A_i \times F(a)_i]$ и $[A_j \times F(a)_j]$ следует

$$F(a_3) = F(a_{10}) = (F_3, F_4, F_5, F_8),$$

но оба элемента участвуют в реализации различных унитарных цветов, поскольку элемент a_3 входит в тела $A_1(F_3)$ и $A(F_5)$, а элемент a_{10} – в тела $A_1(F_4)$ и $A_1(F_8)$, $A_2(F_8)$. Следовательно, на уровне детализации описания (2.66) пересечение этих множеств пусто: $\Pi^{\wedge}S_{ij}^{\circ} = \emptyset$.

Если же рассматривать элементы данных множеств на менее детальном уровне описания (2.64), когда учитываются только цвета $F(a_k)_{F(A)}$, одноименные унитарным цветам $F(A)$, то из матриц $[A_i \times F(A)_i]$ и $[A_j \times F(A)_j]$ следует, что по составам цветов $F(a_k)_{F(A)}$, одноименных унитарным цветам $\Pi^{\wedge}S_i$ и $\Pi^{\wedge}S_j$, будут относительно эквивалентными следующие элементы:

$a_2 \equiv a_7$, так как ;

$a_5 \equiv a_{11}$, так как $F(a_5)_{F(A)} = F(a_{11})_{F(A)} = (F_1, F_3, F_6)$.

Поэтому пересечение $\Pi^{\wedge}S_{ij}^{\circ}$ теперь не будет пустым. Из матрицы $[A_i \times A(F)_i]$ следует, что элементы a_2, a_5 образуют тела $A_2(F_3)$ и $A_1(F_6)$ унитарных цветов $F_3, F_6 \in F(A)_i$; элементы a_7, a_{11} также образуют тела $A_3(F_3)$ и $A_3(F_6)$ унитарных цветов $F_3, F_6(A)_j$. Элементами и унитарными цветами $\Pi^{\wedge}S_{ij}^{\circ}$ будут:

$$A_{ij}^{\circ} = (a_2^*, a_5^*), a_2^* = \{a_2, a_7\}, a_5^* = \{a_5, a_{11}\}; F(A)_{ij} = (F_3, F_6).$$

Булевы матрицы (2.17) и (2.21) пересечения $\Pi^{\wedge}S_{ij}^{\circ}$ показаны на рис.2.5б. В этом пересечении элемент a_2^* является обобщенным представлением двух элементов – $a_2 \in A_i$ и $a_7 \in A_j$; точно так же a_5^* является обобщенным представлением двух элементов – $a_5 \in A_i$ и $a_{11} \in A_j$.

При решении реальных задач вместо a_2^* и a_5^* в составе пересечения будут фигурировать сами элементы исходных $\Pi^{\wedge}S$ -множеств: вместо a_2^* – элемент a_2 или a_7 , вместо a_5^* – элемент a_5 или a_{11} .

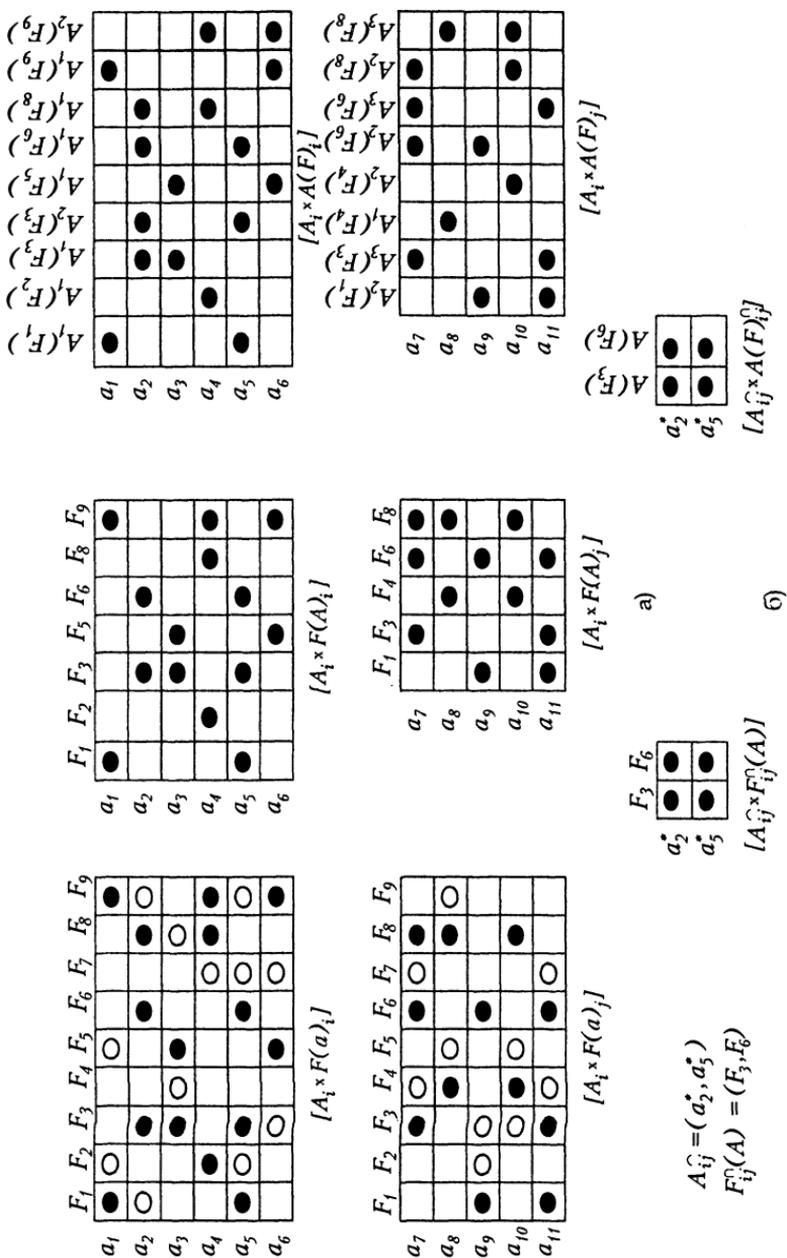


Рис.2.5. Булевы матрицы цветов и тел: а) исходные множества $\Pi^{\wedge} S_i$ и $\Pi^{\wedge} S_j$; б) пересечение $\Pi^{\wedge} S_{ij}$

Разность обычных множеств A_i, A_j вычисляется по формуле:

$$A_{ij}^{\varphi} = A_i \setminus A_j, \quad (2.149)$$

когда

$$\forall a_k \in A_{ij}^{\varphi} [(a_k \in A_i) \wedge (a_k \notin A_j) = 1], \quad (2.150)$$

и по формуле:

$$A_{ij}^{\varphi} = A_j \setminus A_i, \quad (2.151)$$

когда

$$\forall a_k \in A_{ij}^{\varphi} [(a_k \notin A_i) \wedge (a_k \in A_j) = 1]. \quad (2.152)$$

Разность полихроматического множества PS_i и полихроматического множества PS_j

$$PS_{ij}^{\varphi} = PS_i \setminus PS_j \quad (2.153)$$

рассматривается как подмножество компонентов PS_i , оставшихся после исключения из PS_i элементов, эквивалентных элементам множества PS_j :

$$PS_{ij}^{\varphi} = (A_{ij}^{\varphi}, F(a)_{ij}^{\varphi}, F(A)_{ij}^{\varphi}, [A_{ij}^{\varphi} \times F(a)_{ij}^{\varphi}], \\ [A_{ij}^{\varphi} \times F(A)_{ij}^{\varphi}], [A_{ij}^{\varphi} \times A(F)_{ij}^{\varphi}]). \quad (2.154)$$

Поскольку состав A_{ij}^{φ} элементов PS_{ij}^{φ} не более состава A_i элементов PS_i , то состав унитарных цветов $F(A)_{ij}^{\varphi}$ разности PS_i с PS_j не превышает состава унитарных цветов PS_i :

$$F(A)_{ij}^{\varphi} \subseteq F(A)_i. \quad (2.155)$$

Аналогично определяется подмножество PS_{ij}^{φ} элементов PS_j , остающихся после исключения из PS_j элементов, эквивалентных элементам PS_i .

Это позволяет осуществлять определение разности PS_i и PS_j по следующей схеме:

- составы компонентов исходных множеств PS_i и PS_j приводятся к единому уровню детализации их описания;
- описания всех элементов PS_i и PS_j приводятся к нормализованному виду;

- по методике, изложенной выше, вычисляется пересечение PS_{ij}^{\cap} исходных множеств PS_i и PS_j ;
- из исходного множества PS_i исключаются элементы, входящие в пересечение PS_{ij}^{\cap} . Оставшаяся часть множества PS_i и будет искомой разностью PS_{ij}^{\setminus} .

Например, если исходными будут рассмотренные ранее (см. рис.2.5а) множества $P^{\wedge}S_i$ и $P^{\wedge}S_j$, то пересечение элементов этих множеств на уровне детализации описания $\langle a_k, F(a_k)_{F(A)} \rangle$ содержит элементы:

$$A_{ij}^{\cap} = A_{ij}^{\cap} = (a_2^*, a_5^*) = ((a_2 \equiv a_7), (a_5 \equiv a_{11})).$$

Следовательно, при исключении из $P^{\wedge}S_i$ элементов a_2, a_5 , входящих в пересечение A_{ij}^{\cap} , мы получим состав элементов A_{ij}^{\setminus} , входящих в PS_{ij}^{\setminus} :

$$A_{ij}^{\setminus} = A_i \setminus (A_i \cap A_j) = (a_1, a_3, a_4, a_6).$$

После исключения элементов a_2, a_5 в матрице $[A_i \times A(F)_i]$ останутся только тела $A(F_2), A_1(F_5), A_1(F_9)$ и $A_2(F_9)$, поэтому состав унитарных цветов разности $P^{\wedge}S_{ij}^{\setminus}$ равен

$$F(A)_{ij}^{\setminus} = (F_2, F_5, F_9).$$

Булевы матрицы $[A_{ij}^{\setminus} \times F(a)_{ij}^{\setminus}]$, $[A_{ij}^{\setminus} \times F(A)_{ij}^{\setminus}]$ и $[A_{ij}^{\setminus} \times A(F)_{ij}^{\setminus}]$ показаны на рис.2.6а. Аналогично вычисляется разность $P^{\wedge}S_{ij}^{\setminus}$; булевы матрицы разности исходного множества $P^{\wedge}S_j$ и $P^{\wedge}S_i$ (рис.2.5) показаны на рис.2.6б.

Объединение полихроматических множеств PS_i и PS_j рассматривается как новое полихроматическое множество:

$$PS_{ij}^{\cup} = PS_i \cup PS_j; \quad (2.156)$$

состав элементов которого равен A_{ij}^{\cup} :

$$PS_{ij}^{\cup} = (A_{ij}^{\cup}, F(a)_{ij}^{\cup}, (A)_{ij}^{\cup}, [A_{ij}^{\cup} \times F(a)_{ij}^{\cup}], \\ [A_{ij}^{\cup} \times F(A)_{ij}^{\cup}], [A_{ij}^{\cup} \times A(F)_{ij}^{\cup}]). \quad (2.157)$$

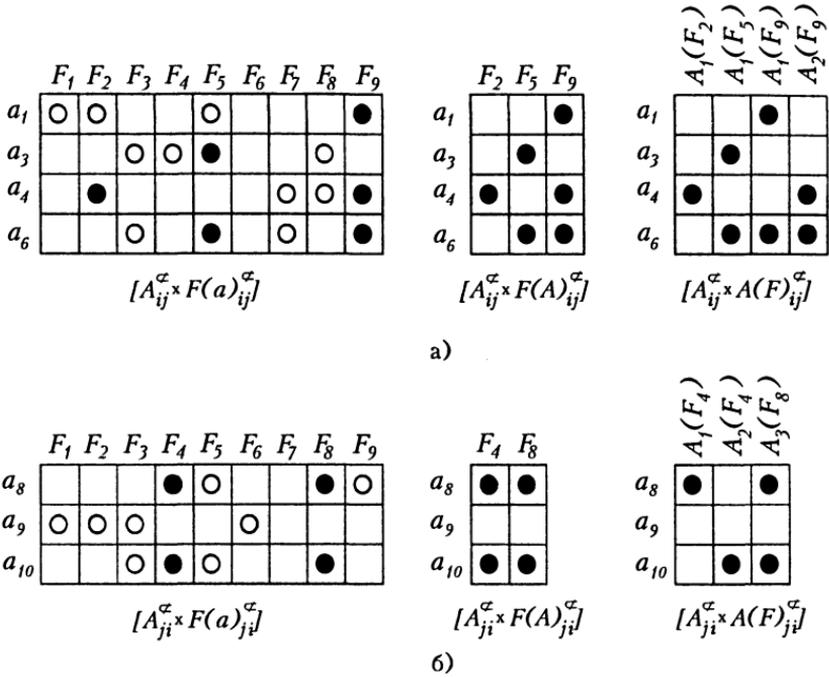


Рис.2.6. Разности множеств $\Pi^{\wedge}S_i$ и $\Pi^{\wedge}S_j$; а) разность $\Pi^{\wedge}S_{ij}^{\sigma}$; б) разность $\Pi^{\wedge}S_{ij}^{\sigma}$

Объединение полихроматических множеств $\Pi S_i, \Pi S_j$ существенно отличается от их пересечения и разности тем, что при объединении конъюнктивных множеств ΠS_i и ΠS_j в составе A_{ij}^{\cup} могут появиться такие совокупности элементов из A_i и A_j , которые порождают новые тела унитарных цветов, в том числе не существовавших ранее ни в $F(A)_i$, ни в $F(A)_j$, поэтому $F(A)_{ij}^{\cup}$ может быть больше объединения $F(A)_i$ и $F(A)_j$. В общем случае их составы персональных цветов элементов и унитарных цветов объединяемых множеств связаны соотношением:

$$(F(a)_i \cup F(a)_j) \supseteq F(A)_{ij}^{\cup} \supseteq (F(A)_i \cup F(A)_j). \tag{2.158}$$

Выявление новых тел и новых унитарных цветов представляет самостоятельную проблему, решаемую методами поиска новых технических решений. Компоненты объединения ΠS_i и ΠS_j определяются по следующей схеме:

- составы компонентов исходных множеств PS_i и PS_j приводятся к единому уровню детализации их описания;
- описания всех элементов PS_i и PS_j приводятся к нормализованному виду;
- вычисляется объединенный состав A_{ij}^{\cup} элементов итогового множества PS_{ij}^{\cup} по формуле $A_{ij}^{\cup} = A_i \cup A_j$;
- если уровень детализации описания исходных дизъюнктивных множеств не ниже (2.63), а конъюнктивных множеств – не ниже (2.66), то по формуле (2.6) вычисляется состав $F(a)_{ij}^{\cup}$ цветов элементов PS_{ij}^{\cup} ;
- формируется булева матрица $[A_{ij}^{\cup} \times A(F)_{ij}^{\cup}]$ тел унитарных цветов PS_{ij}^{\cup} путем объединения матриц $[A_i \times A(F)_i]$ и $[A_j \times A(F)_j]$. В матрицу тел включают и выявленные новые тела унитарных цветов PS_{ij}^{\cup} ;
- по матрице тел определяется состав $F(A)_{ij}^{\cup}$ унитарных цветов PS_{ij}^{\cup} и строится булева матрица $[A_{ij}^{\cup} \times F(A)_{ij}^{\cup}]$.

Поскольку в дизъюнктивных $P^{\vee}S$ -множествах матрицы $[A \times F(A)]$ и $[A \times F(a)]$ совпадают по составу цветов, то операция объединения дизъюнктивных множеств

$$P^{\vee}S = P^{\vee}S_i \cup P^{\vee}S_j$$

может выполняться при представлении исходных множеств в форме (2.69). При объединении дизъюнктивных множеств не могут появиться новые унитарные цвета:

$$P^{\wedge}S_{ij}^{\cup} \rightarrow F(A)_{ij}^{\cup} = F(A)_i \cup F(A)_j.$$

При объединении конъюнктивных множеств

$$P^{\wedge}S_{ij}^{\cup} = P^{\wedge}S_i \cup P^{\wedge}S_j$$

персональные цвета элементов, принадлежащих A_i и A_j , могут порождать за счет образования новых тел новые унитарные цвета, не существовавшие ранее в $F(A)_i$ и $F(A)_j$, поэтому унитарная раскраска объединения связана с унитарными раскрасками исходных $P^{\wedge}S$ -множеств соотношением:

$$P^{\wedge}S_{ij}^{\cup} \rightarrow F(A)_{ij}^{\cup} \supseteq (F(A)_i \cup F(A)_j). \quad (2.160)$$

Построение булевой матрицы $[A_{ij}^{\cup} \times A(F)_{ij}^{\cup}]$ тел унитарных цветов при объединении $P^{\wedge}S_i$ и $P^{\wedge}S_j$ начинается с предварительного объединения булевых матриц $[A_i \times A(F)_i]$ и $[A_j \times A(F)_j]$, а затем в итоговую матрицу включаются новые тела, состоящие из элементов обоих объединяемых множеств (причины появления таких новых тел здесь не рассматриваются). Поэтому матрица тел объединенного множества $P^{\wedge}S_{ij}$ связана с матрицами тел исходных множеств $P^{\wedge}S_i$ и $P^{\wedge}S_j$ соотношением:

$$[A_{ij}^{\cup} \times A(F)_{ij}^{\cup}] \supseteq [A_i \times A(F)_i] \cup [A_j \times A(F)_j]. \tag{2.161}$$

Рассмотрим определение объединения конъюнктивных множеств, при детализации описания их элементов на уровне (2.37), на примере $P^{\wedge}S_i$ и $P^{\wedge}S_j$ (рис.2.5а). Из матриц $[A_i \times F(a)_i]$ и $[A_j \times F(a)_j]$ следует, что в составе элементов исходных $P^{\wedge}S$ -множеств эквивалентны по персональным раскраскам только элементы a_3 и a_{10} , однако в матрицах $[A_i \times F(A)_i]$ и $[A_j \times F(A)_j]$ эти элементы имеют разные составы цветов, одноименных с унитарными цветами $F(A)_i$ и $F(A)_j$. Поэтому элементы a_3, a_{10} также нельзя считать эквивалентными и, следовательно, все элементы исходных $P^{\wedge}S$ -множеств различны. Состав элементов A , вычисляемый по формуле (2.159), при этом равен

$$A_{ij}^{\cup} = A_i \cup A_j = (a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, a_{11}),$$

а состав цветов $F(a)_{ij}^{\cup}$ равен

$$F(a)_{ij}^{\cup} = F(a)_i \cup F(a)_j = (F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8, F_9).$$

Булева матрица цветов элементов $[A_{ij}^{\cup} \times F(a)_{ij}^{\cup}]$ является объединением соответствующих матриц (рис.2.5а), и приведена на рис.2.7а. Состав унитарных цветов $F(A)_{ij}^{\cup}$ и булева матрица $[A_{ij}^{\cup} \times F(A)_{ij}^{\cup}]$ формируются следующим образом. Вначале строится булева матрица тел $[A_{ij}^{\cup} \times A(F)_{ij}^{\cup}]$ и по этой матрице проверяют, не появились ли новые тела унитарных цветов, наряду с приведенными в матрицах $[A_i \times A(F)_i]$ и $[A_j \times A(F)_j]$ на рис.2.5, за счет объединения элементов из $P^{\wedge}S_i$ и $P^{\wedge}S_j$. В рассматриваемом примере такими новыми телами являются

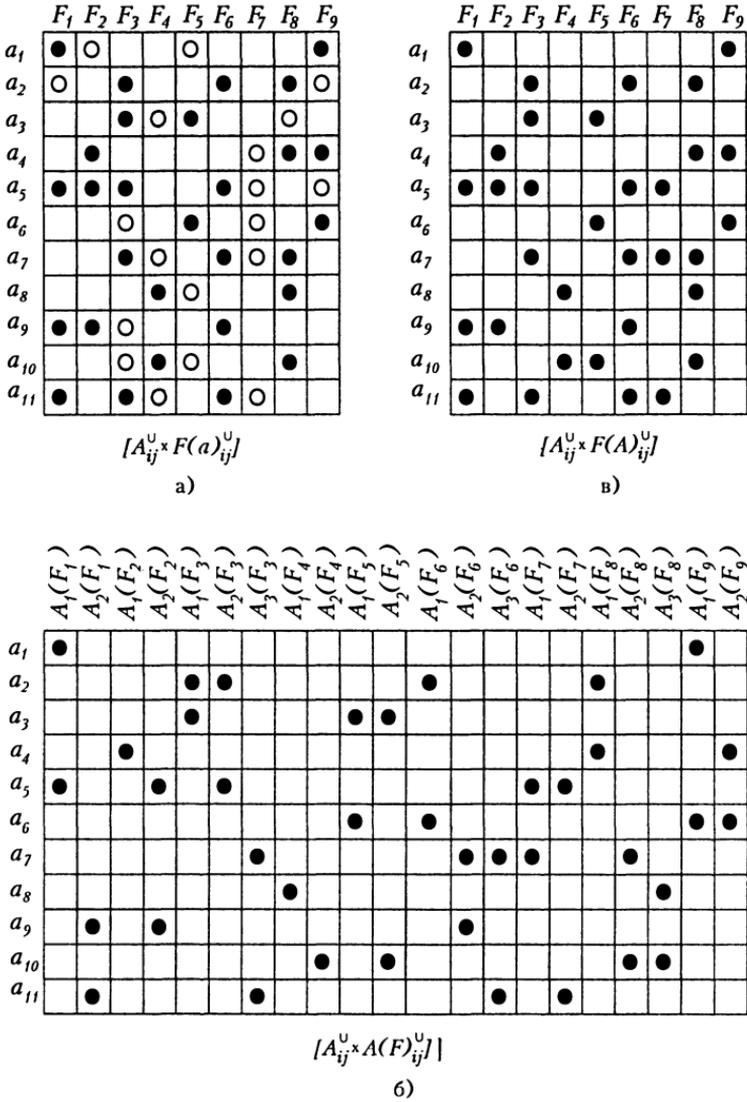


Рис.2.7. Объединение множеств $\Pi^{\wedge}S_i$ и $\Pi^{\wedge}S_j$

$A_2(F_2) = (a_5, a_9), A_2(F_5) = (a_3, a_{10}).$

Кроме того, появились новые тела унитарного цвета $F_7(A)$, не существовавшего ранее в раскрасках исходных $\Pi^{\wedge}S$ -множеств:

$A_1(F_7) = (a_5, a_7), A_2(F_7) = (a_5, a_{11}).$

В результате матрица тел $[A_{ij}^{\cup} \times A(F)_{ij}^{\cup}]$, объединяющая матрицы $[A_i \times A(F)_i]$ и $[A_j \times A(F)_j]$ и новые тела, имеет вид (см. рис.2.7б). На основании состава тел $A(F)_{ij}^{\cup}$ определен состав унитарных цветов

$$F(A)_{ij}^{\cup} = (F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8, F_9)$$

и сформирована булева матрица $[A_{ij}^{\cup} \times F(A)_{ij}^{\cup}]$ (рис.2.7в).

2.1.4. Особенности полихроматических графов

Одним из основных аспектов моделирования сложных систем является отображение различных связей между элементами этих систем, для чего используется аппарат теории графов. Обыкновенным графом $G = (A, C)$ называется упорядоченная пара конечных множеств: множества вершин A , и множества ребер или дуг, соединяющих вершины графа. Однако традиционный математический аппарат теории графов не содержит средств одновременного описания и состава, и разнообразных свойств вершин и ребер графа, что сужает возможности моделирования реальных систем. Такие средства содержит аппарат полихроматических графов, основанный на понятиях теории полихроматических множеств.

Полихроматическим называется граф PG , вершины и/или ребра которого являются полихроматическими множествами. Способы описания полихроматического графа зависят от характера взаимосвязи цветов вершин и ребер по их влиянию на унитарную раскраску $F(G)$ PG -графа. Полихроматические множества вершин и ребер PG -графа имеют вид:

$$PS_A = (A, F(a), F(A), [A \times F(a)], [A \times F(A)], [A \times A(F)]), \quad (2.162)$$

$$PS_C = (C, F(c), F(C), [C \times F(c)], [C \times F(C)], [C \times C(F)]), \quad (2.163)$$

где A, C – множества вершин и ребер, рассматриваемых без учета их раскраски; $F(a), F(c)$ – множества персональных цветов всех вершин и ребер; $F(A), F(C)$ – множества унитарных цветов – унитарные раскраски PS_A и PS_C ; $[A \times F(a)], [C \times F(c)]$ – булевы матрицы персональных раскрасок вершин и ребер; $[A \times F(A)], [C \times F(C)]$ – булевы матрицы цветов вершин и ребер, одноименных унитарным цветам $F(A)$ и $F(C)$; $[A \times A(F)], [C \times C(F)]$ – булевы матрицы вариантов тел, обеспечивающих существование унитарных цветов $F(A)$ и $F(C)$.

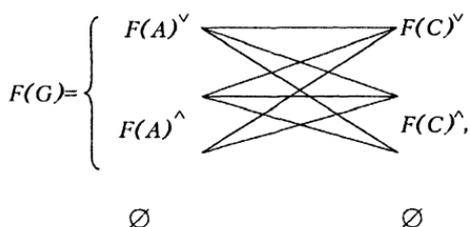
Если условия существования унитарных цветов множества вершин не зависят от условий существования унитарных цветов

множества ребер, то полихроматический граф описывается четверкой компонентов:

$$PG = (G, F(G), PS_A, PS_C), \quad (2.164)$$

где G – описание инцидентности между вершинами и ребрами PG – графа; $F(G)$ – унитарная раскраска PG – графа, определяемая в виде функции, аргументами которой являются раскраски $F(A)$ и $F(C)$ полихроматических множеств PS_A, PS_C . Такой граф называется PS – графом с *независимой раскраской* вершин и ребер. Обычное множество A вершин можно принять как полихроматическое, но с пустым составом унитарных цветов и пустыми составами персональных цветов элементов; точно так же множество ребер C можно принять как полихроматическое, но с пустыми составами унитарных цветов и персональных цветов ребер. Поэтому в любом PG -графе можно выделить множества A и C , определяющие обычный граф $G = (A, C)$, вершины и ребра которого бесцветны; этот граф называется *бесцветным каркасом* PG -графа.

PG -граф представляет собой единый комплекс, объединяющий множества PS_A вершин и PS_C ребер в единое целое. Несмотря на независимость унитарных раскрасок вершин и ребер, при объединении PS_A и PS_C в единый комплекс их унитарные цвета могут быть по-разному взаимосвязаны в унитарной раскраске $F(G)$. Раскраску $F(G)$ PG -графа можно определить как результат теоретико-множественной или логической операции над множествами унитарных цветов $F(A)$ вершин и $F(C)$ ребер по схеме:



где $F(A)^\vee$ или $F(A)^\wedge$ означает, что данная раскраска принадлежит дизъюнктивному или конъюнктивному множеству вершин, а $F(C)^\vee$ или $F(C)^\wedge$ – что данная раскраска принадлежит дизъюнктивному или конъюнктивному множеству ребер PG -графа. Примером PG -графа с дизъюнктивным PS_A – множеством вершин и конъюнктивным PS_C -множеством дуг служит структурная модель производственного участка (рис.2.8). В случае $F(G) = (F(A), \emptyset)$ полихроматический граф имеет вид $PG = (G, PS_A)$, а в случае $F(G) = (\emptyset, F(C))$ имеет вид $PG = (G, PS_C)$.

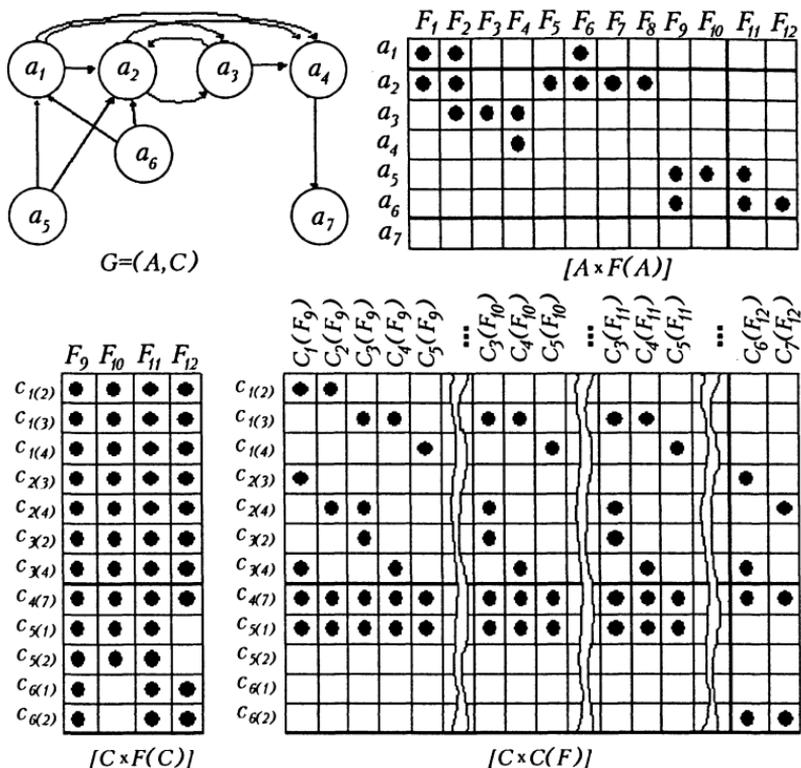


Рис.2.8. PG-граф структурной модели производственного участка.

Вершины PG-графа: a_1 – горизонтально-фрезерный станок;

a_2 – вертикально-фрезерный станок; a_3 – токарный станок; a_4 – сверлильный станок; a_5 – рабочий; a_6 – промышленный робот; a_7 – склад деталей.

Цвета PG-графа (свойства заготовок и деталей): F_1 – плоская поверхность;

F_2 – торец; F_3 – наружная поверхность; F_4 – круглое отверстие; F_5 – фасонная поверхность; F_6 – паз прямолинейный сквозной; F_7 – паз прямолинейный глухой; F_8 – паз криволинейный; F_9 – перемещение заготовки (детали); F_{10} – нежесткая деталь; F_{11} – вес заготовки $P < 15$ кг; F_{12} – вес заготовки $15 < P < 50$ кг

Если в раскраске $F(G)$ PG-графа унитарные цвета $F(A)$ множества $ПС_A$ вершин существуют независимо от унитарных цветов $F(C)$ множества $ПС_C$ ребер, то состав унитарных цветов $F(G)$ выполняется по формуле:

$$F(G) = F(A) \vee F(C), \quad (2.165)$$

а если унитарные цвета $F(A)$ существуют только при наличии одноименных цветов в раскраске $F(C)$ ребер, то

$$F(G) = F(A) \wedge F(C), \quad (2.166)$$

PG -граф с независимой раскраской вершин и ребер используется при моделировании реальных систем в тех случаях, когда элементы системы являются функционально самостоятельными объектами, свойства которых независимы от других объектов и связей между ними. Вершина a_k исключается из состава PS_A вершин такого PG -графа по методике, изложенной выше. В результате получается измененное множество PS_{A_k} вершин вида (2.77). Если в исходном PG -графе существовали ребра, инцидентные исключенной вершине a_k , то эти ребра исключаются из состава PS_C .

При независимой раскраске вершин и ребер добавление в PG -граф новой вершины a_k рассматривается как включение a_k в множество PS_A вершин исходного PG -графа. Введение a_k в PS_A требует также решения вопроса о появлении новых связей между элементами исходного множества вершин и a_k , т.е. о добавлении нового ребра или нескольких ребер.

К основным операциям над PG -графами относятся операции объединения, пересечения и разности PG -графов, а также выделение некоторой части PG -графа. Содержание этих операций различается для графов с независимой и зависимой раскраской вершин и ребер.

Выделение части PG -графа с независимой раскраской вершин и ребер сравнительно простая операция, так как в ней участвуют вершины и ребра единственного, исходного PG -графа, все данные которого известны. В выделяемой части $PG_i \subset PG$ составы унитарных цветов:

$$F(G)_i \subseteq F(G), F(A)_i \subseteq F(A), F(C)_i \subseteq F(C)$$

и условия существования унитарных цветов в $F(G)_i, F(A)_i, F(C)_i$ соответствуют условиям существования этих цветов в исходном PG -графе. $PG_i \subset PG$ выделяется по-разному в зависимости от условий задачи. Если в состав исходных данных входят вершины или ребра искомого части PG -графа, то задача решается двумя путями:

1. В исходном PG -графе задается состав $A_i \subset A$ вершин искомого PG_i -графа, а затем определяется состав $C_i \subset C$ ребер, инцидентные вершины которых входят в A_i , и определяются унитарные цвета $F(A)_i, F(C)_i, F(G)_i$. Способ определения унитарных цветов зависит от условий их существования. В этом случае в PG_i могут существовать изолированные вершины.

2. В исходном PG -графе задается состав $C_j \subset C$ ребер искомого PG_i -графа, а затем определяется состав $A_j \subseteq A$ вершин, инцидентных заданным ребрам. В этом случае в PG_i не будут существовать изолированные вершины.

К операциям выделения части PG -графа относится и построение путей, цепей, циклов и т.п.

Выделение части PG -графа с заданным составом вершин и ребер или дуг используется при решении задач синтеза структуры объекта из известных элементов с последующим определением функциональных свойств этого объекта, представляемых унитарной раскраской выделенной части PG -графа.

В отличие от операции выделения части PG -графа, при вычислении разности, пересечения и объединения участвуют разные PG -графы. Если PG -графы имеют независимую раскраску вершин и ребер, то эти операции могут выполняться раздельно над множествами вершин и ребер исходных PG -графов.

При выполнении операций над разными PG -графами необходимо учитывать условия эквивалентности их вершин и ребер. Следует иметь в виду, что вычисление объединения, пересечения и разности графов PG_i и PG_j сопровождается одновременным выполнением одноименных операций над описаниями инцидентностей G_i, G_j между вершинами и ребрами PG_i и PG_j .

Если раскраски вершин и ребер в PG -графах взаимно независимы по условиям их существования, то все виды операций над PG_i и PG_j выполняются раздельно над их множествами вершин и множествами ребер. При этом следует иметь в виду, что операция над множествами PS_{C_i}, PS_{C_j} ребер должна выполняться после операции над множествами PS_{A_i}, PS_{A_j} вершин, и в состав ребер искомого PG -графа включаются только те ребра, для которых имеются обе инцидентные вершины. После выполнения операций над PG -множествами вершин и ребер определяется унитарная окраска $F(G)$ PG -графа. Выполнение операций пересечения и разности графов PG_i, PG_j не вызывает особых трудностей, поскольку результатом их выполнения являются части исходных PG -графов.

2.1.5. Структурное моделирование производственных систем в CALS-технологиях

Структурные модели описывают состав и взаимосвязь элементов и свойств объекта моделирования. Структурные модели играют важную роль в CALS-технологиях как средство представления основных данных об изделиях и других объектах на всех этапах их жизненного цикла. Эти данные описываются в терминах *сущностей* и *атрибутов* на языке EXPRESS (ISO 10303-11). Математическое моделирование структуры объекта осуществляется средствами аппарата теории множеств, теории графов и математической логики. Для моделирования плохо формализуемых структур используются аппарат реляционной алгебры, семантических сетей и другие специальные средства.

Весьма удобным средством структурного моделирования в CALS-технологиях является система моделирования ИСТРА,

поскольку математический аппарат теории полихроматических множеств и графов естественным образом интерпретирует сущности как объекты обычных множеств и графов, а атрибуты – как цвета в раскрасках *PS*-множеств и *PG*-графов. Тем самым обеспечивается информационный интерфейс между описаниями моделей и регламентированными языковыми средствами CALS-технологий.

Система моделирования ИСТРА представляет собой комплекс методов и средств создания взаимосвязанных моделей, описывающих объекты моделирования на различных уровнях с разной степенью полноты представления данных в зависимости от назначения моделей при автоматизированной обработке информации. Уровни описания объекта соответствуют уровням абстрагирования при моделировании:

- словесное (вербальное, лингвистическое);
- теоретико-множественное;
- логическое;
- количественное.

Вербальный уровень описания объекта отображается в информационном и лингвистическом обеспечении CALS. Теоретико-множественное, логическое и количественное описания входят в состав математического обеспечения CALS, и являются компонентами единой системы моделирования объектов на различных уровнях абстрагирования, обеспечивающей системную связность всех моделей данного объекта и других, функционально и информационно связанных с ним, объектов путем описания формализованных отношений между различными моделями и компонентами этих моделей.

В системе ИСТРА любой объект – изделие или средство технологического оснащения, процесс проектирования или производства, и т.п. – моделируется одинаковыми математическими средствами. Рассматриваемый на любом уровне абстрагирования объект *A* с математической точки зрения имеет один и тот же прообраз, соответствующий денотату реального объекта.

Теоретико-множественный и логический уровни соответствуют этапу структурного проектирования, когда определяется состав и взаимосвязь элементов проектируемого объекта, а количественный уровень соответствует расчету числовых значений параметров и количественной оценке проектного решения. Основным аппаратом структурного моделирования являются математический аппарат теории обычных и полихроматических множеств и графов и аппарат математической логики, а основным аппаратом параметрического моделирования является традиционный аппарат математики, оперирующей с числовыми величинами.

Для решения на теоретико-множественном уровне различных задач в CALS могут использоваться теоретико-множественные операции над PS -множествами. Если при моделировании сложного объекта существенное значение имеют связи между его элементами, то для моделирования на теоретико-множественном уровне используется аппарат теории полихроматических графов.

На логическом уровне все объекты, их элементы и свойства рассматриваются как логические величины, между которыми устанавливаются, при соответствующей интерпретации, логические отношения R^L вида (2.18), (2.29), (2.40) и т.д.

При моделировании на количественном уровне свойствам объекта и его элементов соответствуют количественные величины, имеющие числовое значение, между которыми, при соответствующей интерпретации, устанавливаются количественные отношения R^N .

От одного к другому уровню описания переходят регламентированными способами с помощью межуровневых отношений, с указанием границ и условий перехода одних величин в другие. При переходе на более абстрактный уровень моделирования осуществляется свертка данных о моделируемом объекте, а при переходе к более детальному уровню описания – развертка этих данных. Регламентация способов описания переходов одних величин в другие обеспечивает системную связность описаний теоретико-множественных, логических и количественных свойств и отношений при моделировании объекта.

Система ИСТРА обеспечивает возможность представления в одной математической модели разнородных объектов при переходе к более абстрактным уровням описания, так как некоторые свойства и отношения на более абстрактном уровне оказываются изоморфными. В общем случае моделью $S(A)$ объекта A будет тройка компонентов

$$S(A) = (A, F, R),$$

где A – множество элементов объекта; F – множество контуров (свойств); R – множество отношений между элементами и контурами. В конкретных случаях содержание модели $S(A)$ описывается компонентами полихроматических множеств и графов, цвета в раскрасках которых соответствуют контурам (свойствам) моделируемых объектов, количественными моделями расчета числовых величин, компонентами лингвистического обеспечения и средствами обеспечения связности всех компонентов. Если все эти компоненты представлены в регламентированной форме, то модель будет типовой математической моделью системы ИСТРА. Типовые структурные модели порождающей среды классифицируются в зависимости от структурных свойств самой модели и структурных свойств проектируемого объекта. Отличительной особенностью системы является инвариантность

структурных моделей и алгоритмов проектирования по отношению к смысловому содержанию данных о конкретных объектах проектирования, что обеспечивает возможность применения одних и тех же моделей и алгоритмов для проектирования разнородных объектов, обладающих одинаковыми структурными свойствами.

Состав компонентов структурной модели $S(A)$ различается на разных стадиях проектирования объекта A и зависит от решаемых задач. Так на ранних стадиях концептуального проектирования изделие A может рассматриваться как неделимый объект с определенными характеристиками. Если изделие A рассматривается как сложный объект с составом элементов (2.1), то структурная модель $S(A)$ описывается PS -множеством, детализация описания которого соответствует одному из наборов компонентов (2.60)...(2.66). В простейшем случае структурная модель $S(A)$ состоит из компонентов (2.62)

$$S(A) = (A, F(A)), \quad (2.167)$$

или, если задается бинарное отношение между контурами $F(A)$, то из компонентов (2.63), с добавлением булевой матрицы (2.54):

$$S(A) = (A, F(A), [F(A) \times F(A)]). \quad (2.168)$$

К этим компонентам модели при необходимости добавляются логические отношения (2.51) и (2.53), отражающие условия существования цветов в раскраске $F(A)$.

Аналогичными методами моделируется система проектирования, система производства, система эксплуатации изделия и т.п., а также функциональные и информационные связи между ними, необходимые для реализации CALS-технологии. Так, эффективная деятельность в условиях рынка основана на концепции маркетинга. Технологический мониторинг с учетом концепции маркетинга предполагает, что базисом для достижения целей организации является состояние технологической среды по отношению к производимому изделию. Основными задачами технологического мониторинга производственной системы являются:

- анализ конъюнктуры рынка и конструктивно–технологических свойств существующих и прогнозируемых новых изделий;
- анализ существующих и прогнозирование новых методов и средств производства;
- определение производственных возможностей предприятия и требуемых изменений методов и средств производства с учетом конъюнктуры рынка.

При решении указанных задач используется структурная модель $S(A)$ изделия и структурная модель $S(P)$ производственной системы P . Для приближенной оценки возможности реализации производства

изделия A_i с составом контуров $F(A_i)$ в производственной системе P используется обобщенный булев вектор $F(P)$, описывающий производственные возможности системы в пространстве контуров, охватывающей свойства и системы P , и производимого изделия A_i . Для описания взаимосвязи моделей изделия и производственной системы используется дизъюнктивная форма связи контуров, при которой истинностные значения одноименных контуров изделия и производственной системы имеют следующее смысловое содержание:

$$F_k(A_i) = \begin{cases} 1, & \text{если контур } F_k \text{ изделия реализован;} \\ 0 & \text{– в противном случае;} \end{cases} \quad (2.169)$$

$$F_k(P) = \begin{cases} 1, & \text{если производственная система } P \\ & \text{реализует контур } F_k(A_i) \text{ изделия;} \\ 0 & \text{– в противном случае.} \end{cases} \quad (2.170)$$

Состояние контуров $F(A)_P$ изделия A_i после воздействия производственной системы P определяется по формуле :

$$F(A_i)_P = F(A_i)_O \vee F(P), \quad (2.171)$$

где $F(A_i)_O$ – исходное состояние контуров изделия (материалов, полуфабрикатов и т.п.) перед началом производства. Множество контуров, реализуемых в $F(A_i)_P$, может превышать состав $F(A_i)$ требуемых контуров изделия, так как производственные возможности системы P могут быть избыточными по отношению к составу контуров $F(A_i)$. Производственные возможности системы P анализируются с использованием формул, по смысловому содержанию аналогичных формулам (2.124)–(2.127), которые здесь интерпретируются следующим образом:

– группа контуров изделия A_i , не реализованных в $F(A_i)_O$, которые могут быть реализованы в производственной системе P , вычисляется по формуле:

$$F(A_i)_P^I = \bar{F}(A_i)_O \wedge F(P), \quad (2.172)$$

где $\bar{F}(A_i)_O$ – инверсия исходного булева вектора $F(A_i)_O$; если следует рассматривать состояние только требуемых контуров $F(A_i)$, то вместо (2.172) используется формула

$$F(A_i)_P^I = \bar{F}(A_i)_O \wedge F(P) \wedge F(A_i); \quad (2.173)$$

– группа контуров изделия, реализованных в исходном состоянии булева вектора $F(A_i)_O$, которые могли бы быть реализованы в системе P , вычисляется по формуле:

$$F(A_i)_P^{II} = F(A_i)_O \wedge F(P); \quad (2.174)$$

– группа контуров изделия, реализованных в исходном состоянии булева вектора $F(A_i)_O$, которые не могут быть реализованы в системе P , вычисляются по формуле:

$$F(A_i)_P^{III} = F(A_i)_O \wedge \bar{F}(P), \quad (2.175)$$

где $\bar{F}(P)$ – инверсия булева вектора $F(P)$;

– группа требуемых контуров изделия, не реализованных в исходном состоянии булева вектора $F(A_i)_O$, и нереализуемых в производственной системе P , вычисляется по формуле:

$$F(A_i)_P^{IV} = \bar{F}(A_i)_O \wedge \bar{F}(P) \wedge F(A_i). \quad (2.176)$$

Изделие может быть изготовлено на данном предприятии при условии

$$F(A_i) = (F(A_i)_O \vee F(A_i)_P^I) \vee F(A_i); \quad (2.177)$$

в этом случае, очевидно, $F(A_i)_P^{III} = 0$. В случае, когда условие (2.183) не выполняется, группа контуров $F(A_i)_P^{IV} \neq 0$ останется не реализованной, и для их реализации требуется либо дополнить производственную систему новыми средствами производства, либо найти подрядчика на выполнение работ, связанных с реализацией этих контуров изделия. Если необходимые средства производства отсутствуют и на предприятии, и у подрядчиков, и на рынке, то они должны быть созданы заново; при этом информация о контурах $F(A_i)_P^{IV}$ войдет в техническое задание на проектирование таких средств.

Для более детального анализа производственных возможностей система P декомпозируется до уровня производственных подразделений и средств оснащения производства. В этом случае разрабатывается структурная модель $S(P)$ производственной системы, представляемая PS -множеством или PS -графом на более детальных уровнях их описания с добавлением количественных моделей для расчета технико-экономических показателей. В эту модель включаются как те средства производства, которыми предприятие располагает, так и те, которые могут быть приобретены. Включение последних в модель $S(P)$ позволяет выявить роль и эффективность этих средств до того, как они будут приобретены, что открывает возможность обосновать целесообразность их приобретения для включения в производственную систему.

Аналогичными методами создаются структурные модели для решения других задач технологического проектирования.

2.2. Моделирование технологических процессов

2.2.1. Общие положения

Моделирование технологических процессов является составной частью проведения фундаментальных исследований и прикладных разработок в области конструкторско-технологической информатики – создание, представление и использование знаний о новых технологиях и производственных системах с целью построения методологии и теории проектирования интегрированных автоматизированных производств, поддерживающих полный жизненный цикл изделий машиностроения.

Рассматриваемые задачи и составляют содержательное ядро компьютеризованных информационных систем, охватывающих все этапы жизненного цикла изделия. Информационные системы такого типа получили за рубежом название CALS (continuous acquisition life-cycle support), а процессы их функционирования – CALS-технологий.

Основные направления научных исследований:

- конструирование и технология, включающие концептуальное проектирование изделий, анализ действующих технологий, разработку принципиально новых технологических процессов машиностроительных производств, проектирование оптимальной технологической среды, разработку экспертной системы технологий, разработку прикладных конструкторско-технологических САПР;
- организация, планирование и управление, включающие анализ действующих управленческих структур, концептуальное проектирование интегрированных автоматизированных систем управления, проектирование комплекса программно-аппаратных средств.

Исключая несколько специфические задачи маркетинга, конструкторско-технологическая информатика сводится к решению следующих задач: конструирование изделия, проектирование технологических процессов его изготовления, создание или выбор технологического оборудования, организация и управление технологическими и производственными процессами.

Рассмотрим подробно задачу проектирования технологического процесса изготовления детали изделия машиностроения.

Курс технологии машиностроения рассматривает проектирование технологического процесса в содержательных терминах. Один из возможных вариантов следующий. Весь процесс проектирования разбивается на ряд этапов. Этап – законченная часть процесса проектирования, на которой принимается какое-либо решение.

Общие исходные данные для проектирования.

1. Чертеж детали.
2. Годовая программа и общий выпуск (размер заказа и сроки).
3. Сведения об оборудовании, на котором предполагается изготавливать деталь.

Этап 1. Выбор заготовки.

Например, для изготовления заготовки корпусной детали могут быть применены следующие методы: литье в кокиль, литье под давлением, в оболочку, в песчаную форму, т.е. дано $MMЗ = \{MЗ_i\}$, $i = 1, m$ – множество методов получения заготовки.

Цель выбора $C_з + C_м \rightarrow \min$, где $C_з$ – затраты на получение заготовки, $C_м$ – затраты на механическую обработку.

Возникает противоречие: чем меньше $C_з$, тем больше $C_м$, и наоборот. Решить задачу нельзя, так как на этом этапе $C_м = f(C_з)$ – неизвестна. На этапе 1 имеем множество вариантов.

Этап 2. Выбор планов обработки поверхностей.

Дано: $MPов. = \{Пов. j\}$, $j = 1, n$ – множество поверхностей детали, подлежащие механической обработке.

Получить

$$Пов. j \xrightarrow{f_1} \langle mo_1^j \prec mo_2^j \prec \dots \prec mo_p^j \rangle = \\ = \langle MO^j, R_j \rangle, MO^j = \{mo_k^j\}, k = \overline{1, p},$$

где mo^j – метод обработки j -й поверхности; MO^j – множество методов обработки j -й поверхности; R_j – отношение порядка на множестве MO .

Таким образом, план обработки поверхности есть пара (отношение) $\langle MO^j, R_j \rangle$.

Этап 3. Выбор варианта базирования.

При назначении планов обработки поверхностей детали каждой поверхности ставится в соответствие упорядоченное множество поверхностей, называемое технологическим комплексом.

$$Пов. j \xrightarrow{f_1} \langle пов. j_0 \prec пов. j_1 \prec \dots \prec пов. j_n \prec \dots \prec пов. j_p \rangle$$

$k = 0, p$; j_0 – принадлежит заготовке; j_p – готовой детали; j_0, j_1, j_2, \dots – технологический комплекс.

Каждой поверхности (за исключением j_0) в комплексе ставится в соответствие поверхность, называемая технологической базой:

$$пов. ik \xrightarrow{f_2} пов. jr, j = \overline{1, n}; k = \overline{1, p}; l \in n; r \in k_l.$$

На этом этапе также возможны варианты.

Критерий выбора $C_3 + C_m \rightarrow \min$.

Совокупность размеров от базы до обработанной поверхности + размеры заготовки + размеры готовой детали образуют размерную структуру технологического процесса.

Размеры готовой детали заданы на чертеже. Размеры от базы до обрабатываемой поверхности называются технологическими размерами. Эти размеры колеблются в пределах $A_{\max}^T - A_{\min}^T = \omega^T$ – погрешность технологического размера, $\omega^T = f(V, S, Z)$. Здесь V, S, Z – параметры режима обработки: скорость резания, подача, припуск. Поэтому возникает задача моделирования – нахождения зависимости f и задача управления, т.е. нахождение таких V и S для каждой обработки, чтобы $C_3 + C_m \rightarrow \min$.

Для решения перечисленных задач требуются следующие математические модели.

1. Таблицы принятия решений – планы обработки.
2. Графы – размерные структуры.
3. Непрерывные статистические модели.
4. Оптимизационные линейные и нелинейные задачи.
5. Перебор – комбинаторика.

Резюмируя изложенное, можно предложить некоторую последовательность (алгоритм) проектирования технологического процесса.

1. Генерация методов получения заготовки для заданной детали (эвристика).

2. Генерация вариантов базирования – комбинаторная задача на графах, сводящаяся к перечислению всех корневых деревьев, покрывающих заданное количество вершин.

3. Множество элементов декартова произведения множеств, полученных на первых двух этапах, позволяет построить множество размерных структур технологического процесса.

Комментарий. Каждая размерная структура обладает параметрами (размерами): весом ребер графа. Невозможность варьирования целевой функции одновременно на структурах и параметрах требует вначале параметрической оптимизации каждой структуры, а затем перебор структур.

Параметрическая оптимизация сводится к определению такого набора управляющих параметров (режимов работы технологического оборудования), чтобы доставить экстремум целевой функции, например минимизация всех видов затрат.

Технологическая система (ТС) рассматривается как управляемый объект, обладающий входами и выходами, со следующей классификацией переменных (рис.2.9).

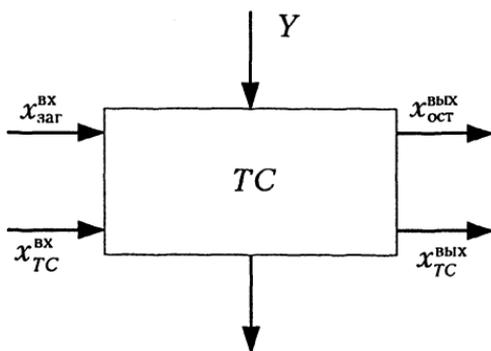


Рис.2.9. Технологическая система

$x_{заг}^{вх}$ – переменные, характеризующие заготовку, поступающую на обработку;

$x_{ТС}^{вх}$ – переменные, характеризующие технологическую систему перед обработкой;

$x_{дет}^{вых}$ – переменные, характеризующие деталь после обработки;

$x_{ТС}^{вых}$ – переменные (показатели) процесса обработки;

Y – управляющие переменные;

θ – целевая функция (критерий оптимальности).

Задача моделирования состоит в формировании критерия оптимальности и в установлении связей между выходными переменными, входными и управляющими.

Моделирование собственно технологической системы основано, как правило, на статистических методах, в частности применяется планирование экспериментов в комбинации с аналитическими методами (термо-упруго-пластические задачи).

Оптимизация либо статическая (временной фактор не играет большой роли), либо динамическая.

В силу большой размерности, нелинейности, большого количества ограничений при статической оптимизации применяется, например, ЛП-поиск. Динамическая оптимизация – динамическое программирование или вариационное исчисление.

5. Перебор параметрически оптимизированных структур и нахождение оптимальной.

Спроектированный до уровня размерной структуры и последовательности обработки отдельных поверхностей технологический процесс необходимо реализовать на соответствующем технологическом оборудовании. Предварительный выбор оборудования был сделан

при определении методов обработки. Но следует учитывать, что на производстве одновременно изготавливают множество деталей. Отсюда возникает последовательность организационно-технологических задач: определение минимального количества технологического оборудования – задача целочисленного программирования; определение минимального количества инструментов и т.п.

2.2.2. Основные математические понятия

M – множество.

m – элемент множества.

$m \in M$ – принадлежность.

$m \notin M$ – непринадлежность.

Множество задается:

1. Перечислением элементов: $M = \{m_i\}, i = \overline{1, k}$.

2. Условиями: $A = \{a / F_1, F_2, \dots\}; F_1, F_2, \dots$ – условия.

$$A = \{a = C / R_e C = 0\},$$

C – множество комплексных чисел, a – чисто мнимые числа.

Операции над множествами.

Объединение

$$A = M \cup N; \langle \cup \rangle - \text{объединение.}$$

$$A = \{a / a \in M \vee a \in N\}; \langle \vee \rangle - \text{или.}$$

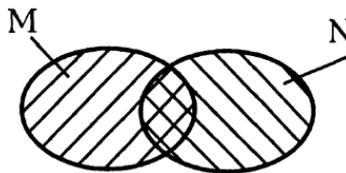
\vee (или) – логическая операция задается таблицей истинности (см. табл.2.)

Например, X, Y, Z – логические переменные, припимающие значения 1 и 0.

Таблица 2

| X | Y | Z = X ∨ Y |
|---|---|-----------|
| 1 | 1 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 0 | 0 | 0 |

Изображение на кругах Венна



$$A = M \cup N; \langle \cup \rangle \text{ или}$$

Пересечение

$$A = M \cap N; A = \{a / a \in M \wedge a \in N\}$$

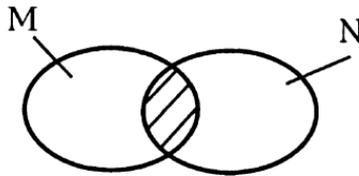
« \cap » – пересечение.

« \wedge » – обозначает букву «и».

\wedge (и) – логическая операция задается таблицей истинности (см. табл.3).

Таблица 3

| X | Y | Z = X \wedge Y |
|---|---|------------------|
| 1 | 1 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 0 | 0 | 0 |



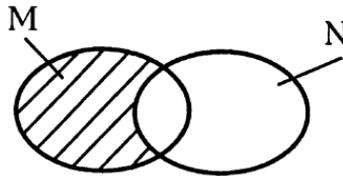
$$A = M \cap N$$

Разность множеств

$$A = M \setminus N, A = \{a / a \in M \wedge a \notin N\}$$

« \setminus » – разность.

« \setminus » – обозначает букву «и».



$$A = M \setminus N$$

Подмножества

$M \subset N$ – «M» является подмножеством «N».

$$a \in M \Rightarrow a \in N$$

Декартовы произведения множества

$$A = M \times N, A = \{a / a = t, n \setminus t \in M, n \in N\} \text{ (см. рис.2.10).}$$

Примечание. Имеет значение порядок сомножителей.

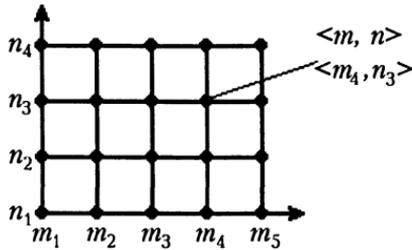


Рис.2.10. Изображение на решетке декартова произведения

Примеры:

$R \subset M \times N$, R – действительная прямая – множества точек прямой.

$R^2 = R \times R$ – плоскость,

$M \times M = M^2$ – вторая степень декартова произведения,

$M \times M \times \dots \times M = M^n$ – n -я степень декартова произведения,

$A \times B \times C \times D \times \dots$ – декартово произведение разных множеств.

Логическое следствие

$$M \subset N, N \subset N \Leftrightarrow M = N$$

Отношения

$R \subset M \times N$ – называется отношением на M , или

$$m_i R m_j \Leftrightarrow \langle m_i, m_j \rangle \in R \subset M \times M.$$

Отношением будем называть подмножество декартова произведения. Если два множества, то отношение – бинарное, если три – тернарное. В основном будем рассматривать бинарные отношения вида $R \subset M^2$.

$R \subset M^n$, n -арное произведение.

$R = \langle M, R \rangle$, $R \subset M^2$, M – носитель отношения, R – сигнатура отношения.

$m_i R m_j$ « m_i » находится в отношении « R » с элементом m_j

С понятием отношение тесно связано понятие – «граф» $G \subset M \times M$. Графом G называется подмножество декартового произведения $M \times M$, то что изображено на решетке (см. рис.2.11.). Элементы множества M будем изображать «0», m_i будем называть вершинами графа (рис.2.12.)

Линия, соединяющая две вершины, называется *дугой* графа или *ребром*.

$m_1 - m_5 \rightarrow$ вершины графа.

Отношение, или граф, носит также название структуры.

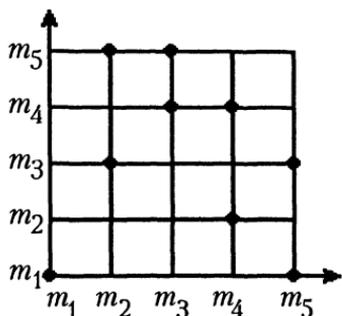


Рис.2.11. Изображение на решетке отношения

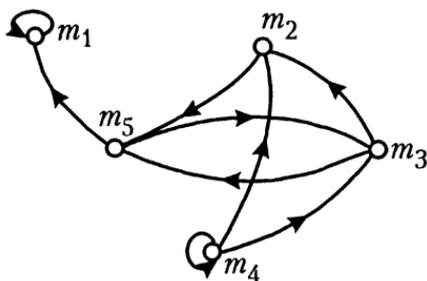


Рис.2.12. Графическое изображение – диаграмма графа

Примеры структур

Деталь как структура (рис.2.13.)

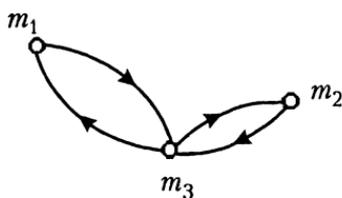
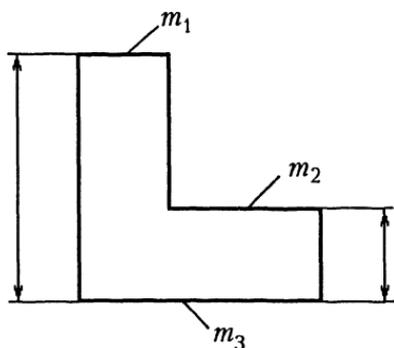


Рис.2.13. Деталь и соответствующий ей граф

Деталь может быть представлена как пара, т.е.

$$D = M, R,$$

где D – деталь, M – множество поверхностей, R – отношение размерной связности поверхностей.

Некоторые аспекты детали можно моделировать при помощи структуры (или отношения).

Технологический процесс как структура

Рассмотрим технологический процесс изготовления детали.

Определение: технологический процесс – это часть производственного процесса, содержащая действия по изменению или определению состояния заготовки или полуфабриката.

Технологический процесс (ТП) можно представить как последовательность операций O_i .

$$ТП = \langle O_1 \prec O_2 \prec \dots \prec O_i \prec \dots \prec O_m \rangle = \langle O, R_0 \rangle$$

$O = \{O_i\}$, $i = 1, m$, R_0 – порядок следования операции.

Операция – это законченная часть технологического процесса, выполняемая на одном рабочем месте и на одном технологическом оборудовании.

Операции в ТП *упорядочены*.

Операция состоит из установов.

Установ – это часть операции, характеризующаяся неизменным положением детали.

$$O_i = \langle Y_{i1} \prec Y_{i2} \prec \dots \prec Y_{ij} \prec \dots \prec Y_{in} \rangle = \langle Y_i, R_y \rangle$$

$$Y_i = \{Y_{ij}\}, j = 1, n$$

Y – установ, « \prec » – знак предшествование, Y_{ij} – j -й установ i -й операции.

В свою очередь, установы ущем инструментом.

$$Y_{ij} = \langle \Pi_{ij1} \prec \Pi_{ij1} \prec \dots \prec \Pi_{ijk} \prec \dots \prec \Pi_{ijp} \rangle = \langle \Pi_{ij}; R_{\Pi} \rangle,$$

$$\Pi_{ij} = \{\Pi_{ijk}\}, k = \overline{1, p}.$$

Π_{ij1} – первый переход j -го установа i -й операции.

Переходы также упорядочены. Переход состоит из рабочих ходов.

Π – переход, R – отношение порядка следования, Π_{ijk} – k -ый переход j -го установа i -й операции – состоит из рабочих ходов.

Рабочий ход (PX) – часть перехода, характеризующаяся однократным перемещением режущего инструмента относительно обрабатываемой поверхности.

$$\Pi_{ijk} = \langle PX_{ijk1} \prec PX_{ijk2} \prec \dots \prec PX_{ijkl} \prec \dots \prec PX_{ijk r} \rangle = \langle PX_{ijk}, R_{PX} \rangle,$$

$$PX = \{PX_{ikl}\}, l = \overline{1, r}.$$

Таким образом, ТП является *структурой* (рис.2.14).

Такого рода структуры носят название *иерархических структур*.

Как правило, в иерархических системах физический смысл имеет только элементы самого нижнего уровня. Высшие уровни носят организационный или информационный характер.

Изделие (сборочная единица) как структура (рис.2.15).

В данном случае имеем пять уровней. Если элементом каждого уровня поставить в соответствие вершины графа, то свяжем – ребра (рис.2.16).

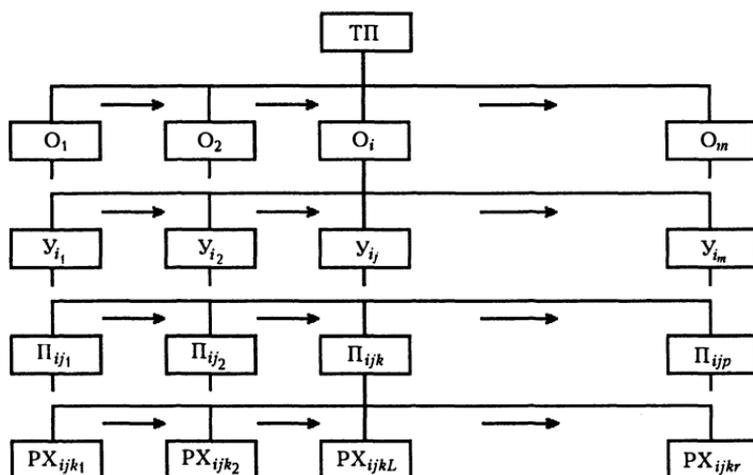


Рис.2.14. Структура технологического процесса

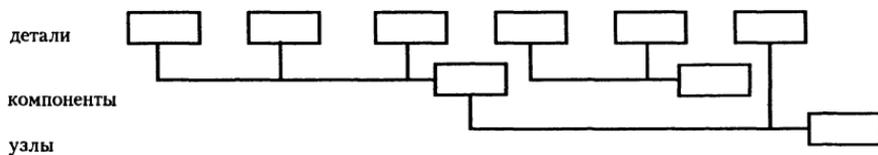


Рис.2.15. Схема сборки

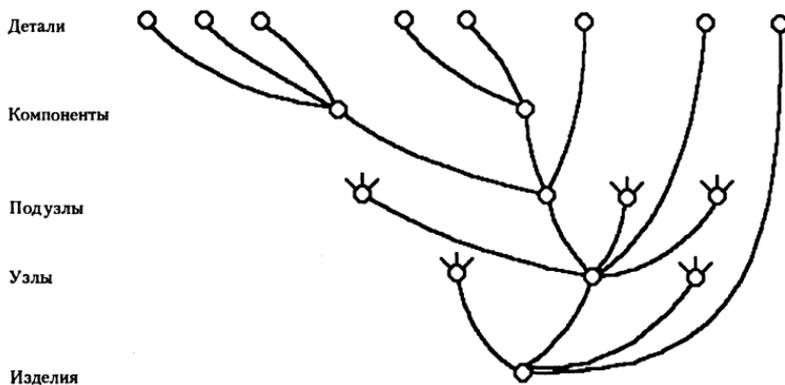


Рис.2.16. Представление схемы сборки графом

2.2.3. Таблицы принятия решений (таблицы соответствий)

Табличные модели весьма распространены в науке и технике. Во многих задачах проектирования мы имеем дело с

1. множеством решений (вариантов).
2. множеством условий, влияющих на выбор решения.

Каждое условие в свою очередь принимает некоторое множество значений. Количество независимых условий, влияющих на выбор решений, может быть достаточно велико. Количество значений каждого условия также может достигать больших величин. Таким образом, общее количество значений условий, равное прямому (декартову) произведению множеств независимых условий, достигает весьма больших величин.

Разработан специальный метод решения таких задач, названный методом поиска решений с помощью таблиц соответствий.

Примеры:

Необходимо выбрать:

1) план обработки поверхности: $Y = \text{План обработки} = f(X_1 - \text{вид поверхности}, X_2 - \text{точность обработки}, X_3 - \text{шероховатость}, X_4 - \text{габариты поверхности})$;

2) сборочный робот: $Y = \text{Робот} = f(X_1 - \text{вид движения}, X_2 - \text{точность движения}, X_3 - \text{вес детали})$;

3) гидропривод: $Y = \text{Гидропривод} = f(X_1 - \text{max скорость}, X_2 - \text{min скорость}, X_3 - \text{усилие}, X_4 - \text{тип гидросистемы})$.

Y – множество решений.

$Y = \{y_l\}, l = 1, k$.

X – условия.

$X = \{X_1, X_2, X_j, X_m\}, j = 1, m$.

$X = \{X_1 \times X_2 \times \dots \times X_j \times \dots \times X_m$

$X = \{X_{ji}\}, i_j = \overline{1, n_j}, j = \overline{1, m}, n_j - \text{количество значений } j\text{-го условия.}$

Соответствием называется следующий кортеж:

$\Gamma = \langle G, Y, X \rangle, G \subset Y \times X$,

Γ – соответствие, G – подмножество $Y \times X$ – график соответствия (см. рис.2.17).

$Y = f(x)$ или $y = f(x)$ – функциональная зависимость (см. рис.2.18).

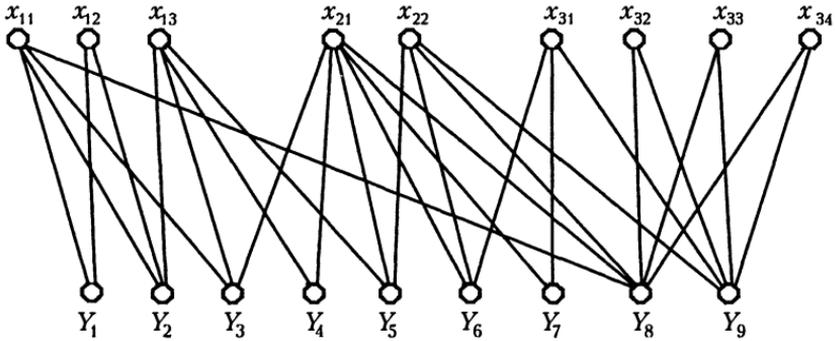


Рис.2.17. График соответствия между значениями аргумента и значениями функции

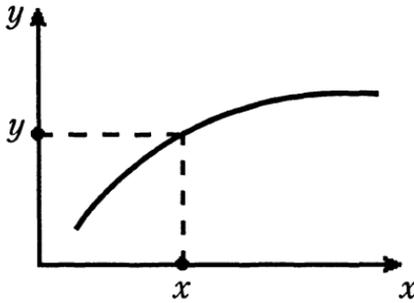


Рис.2.18. График функции

Функция устанавливает соответствие между значением аргумента и значением функции. График функции – множество пар: график $= \{y, x\}$.

Область определения X – декартово произведение непересекающихся независимых множеств.

$$X = X_1 \times X_2 \times \dots \times X_m, X_j = \{x_{j1}, x_{j2}, \dots, x_{jij}, \dots, x_{jmj}\}.$$

Каждое условие принимает n_j значений, тогда X – множество m -оп вида,

$$X = \left\{ \langle x_{1i1}, x_{2i2}, \dots, x_{jij}, \dots, x_{mim} \rangle \right\}, i = \overline{1, n_j}, j = \overline{1, m}.$$

В терминах соответствий: X – область отправления, Y – область прибытия соответствия – множество возможных решений (см. рис.2.18.)

Постановка задачи

Задан некоторый кортеж:

$$\alpha_p = \langle x_{1i1}, x_{2i2}, \dots, x_{jij}, \dots, x_{mim} \rangle.$$

Требуется найти Y^* -образ, т.е. те решения, которые могут приниматься при данном наборе условий.

Зададим связь между значениями условий и решениями в виде некоторого графа:

Например, задан конкретный кортеж:

$\alpha_p = \langle X_{13}, X_{21}, X_{32} \rangle$ – выбор решения зависит, от третьего значения первого условия, от первого значения второго условия и от второго значения третьего условия.

$$\alpha_p = \langle X_{13}, X_{21}, X_{32} \rangle \rightarrow Y^*, Y^* - \text{образ кортежа (решение)}.$$

$$Y(X_{13}) = \{y_2, y_3, y_5\}; Y(X_{21}) = \{y_2, y_4, y_5\};$$

$$Y(X_{32}) = \{y_5, y_8, y_9\}; Y^* = Y(X_{13}) \cap Y(X_{21}) \cap Y(X_{32}) = y_5.$$

Возможны следующие варианты.

1. Y^* – может быть одноэлементным множеством \rightarrow тогда решение единственно.

2. Y^* – многоэлементное множество \rightarrow тогда имеем несколько вариантов решений.

3. Y^* – пустое множество \rightarrow нет решений.

Это говорит о том, что заданная система значений и условий несовместима с заданным множеством решений (см. табл.4).

Таблица 4

Форма задания соответствия

| Множество решений | условие X_1 | | | условие X_2 | | условие X_3 | | |
|-------------------|---------------|----------|----------|---------------|----------|---------------|----------|----------|
| | X_{11} | X_{12} | X_{13} | X_{21} | X_{22} | X_{31} | X_{32} | X_{33} |
| | | | | | | | | |

Порядок заполнения таблицы принятия решений.

1. На основе анализа предметной задачи формулируется множество решений

$Y = \{y_l\}, l = \overline{1, k}$ и записывается в левый столбец таблицы соответствия.

2. На основе анализа предметной задачи формулируется множество условий, влияющих на решение

, т.е. всего у нас m условий, влияющих на решения.

3. На основе анализа предметной задачи для каждого условия определяются множества значений этого условия.

$$X_j = \{X_{ji}\}, i_j = \overline{1, n_j}.$$

4. Множество возможных решений $Y = \{y_i\}$, $l = \overline{1, k}$ и множество возможных значений условий $X_i = \{X_{ji}\}$, $i_j = \overline{1, n_j}$ определяют матрицу соответствий размерностью: $K \times \sum_{j=1}^m \sum_{i_j=1}^{n_j} i_j$.

5. Для каждого решения y_e в l -й строке таблицы ставят единицу в клетках, соответствующих тем значениям условий, для которых данное решение существует.

Таким образом, процесс заполнения таблицы (матрицы) соответствий разбивается на ряд элементарных шагов, на каждом рассматривается только одно решение и только одно значение условия.

Анализ и корректировка таблиц

Таблица 4 является предварительной, ее надо проанализировать и скорректировать. Пусть задана таблица:

$$Y = \{y_i\}, l = \overline{1, k}, X = \{x_{jij}\}, j = \overline{1, m}, i_j = \overline{1, n_j}.$$

И матрица размерностью:

$$K \times \sum_{j=1}^m n_j.$$

Требуется построить граф-схему анализа и поиска решений. Множеству вершин графа G ставится в соответствие множество условий:

$$X = \{X_j\}, j = \overline{1, m}.$$

$$x_{jij}, i_j = \overline{1, n_j}, j = \overline{1, m}.$$

Множеству ребер – множество значений условий.

На вершины графа G навешиваются также решения y_l . (Граф G определяется как множество ребер).

Вначале, перед построением графа G , составляется нормализованная таблица, т.е. из множества линейно-зависимых столбцов оставляется один, удаляются единичные столбцы.

Вводится понятие коэффициента плотности. Это число a_j , равное количеству единиц в столбцах, относящихся к j -му условию:

$$a_j = \sum_{i=1}^k \sum_{i_j}^{n_j} x_{ij}.$$

Таблица упорядочивается в порядке возрастания плотности соответствия (см. табл.5, 6).

Таблица 5

Исходная

| Y | X ₁ | | | | X ₂ | | X ₃ | | | | |
|----------------|---------------------|-----------------|-----------------|-----------------|--------------------|-----------------|---------------------|-----------------|-----------------|-----------------|-----------------|
| | X ₁₁ | X ₁₂ | X ₁₃ | X ₁₄ | X ₂₁ | X ₂₂ | X ₃₁ | X ₃₂ | X ₃₃ | X ₃₄ | X ₃₅ |
| y ₁ | 1 | | 1 | 1 | 1 | | 1 | | 1 | 1 | |
| y ₂ | 1 | 1 | 1 | | 1 | 1 | 1 | 1 | 1 | | 1 |
| y ₃ | | 1 | 1 | 1 | | 1 | | | | 1 | |
| y ₄ | 1 | | 1 | | 1 | | 1 | 1 | 1 | | 1 |
| y ₅ | | 1 | 1 | 1 | 1 | 1 | | | | 1 | |
| y ₆ | 1 | 1 | 1 | | | 1 | 1 | | 1 | | 1 |
| y ₇ | 1 | | 1 | 1 | 1 | | 1 | 1 | 1 | 1 | 1 |
| | a ₁ = 13 | | | | a ₂ = 9 | | a ₃ = 16 | | | | |

Таблица 6

Преобразованная

| Y | X ₁ | | X ₂ | | | X ₃ | | | |
|----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | X ₁₁ | X ₁₂ | X ₂₁ | X ₂₂ | X ₂₃ | X ₃₁ | X ₃₂ | X ₃₃ | X ₃₄ |
| y ₁ | 1 | | 1 | | 1 | 1 | | 1 | |
| y ₂ | 1 | 1 | 1 | 1 | | 1 | 1 | | 1 |
| y ₃ | | 1 | | 1 | 1 | | | 1 | |
| y ₄ | 1 | | 1 | | | 1 | 1 | | 1 |
| y ₅ | 1 | 1 | | 1 | 1 | | | 1 | |
| y ₆ | | 1 | 1 | 1 | | 1 | | | 1 |
| y ₇ | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |

Для построения графа G по таблице соответствий вводится понятие яруса. J -ым ярусом графа G называется совокупность вершин с выходящими из них дугами, длина пути к которым из начальной вершины равна $j - 1$.

Первый ярус

$$G_1 = \langle X_1, \{x_{1/1}\} \rangle, i_1 = \overline{1, n_1}$$

Второй ярус

$$G_2 = \langle x_{11}, \{x_{2/2}\}; x_{12}, \{x_{2/2}\}; x_{13}, \{x_{2/2}\}, \dots, x_{1/1}, \{x_{2/2}\}, \dots \rangle, i_2 = \overline{1, n_2}$$

j -й ярус

$$G_j = \langle x_{j-1,1}, \{x_{jij}\}; x_{j-1,2}, \{x_{jij}\}, \dots, x_{j-1,ij-1}, \{x_{jij}\}, \dots \rangle, i_j = \overline{1, n_j}$$

m -й ярус

$$G_m = \langle x_{m-1,1}, \{x_{mim}\}; x_{m-1,2}, \{x_{mim}\}, \dots, x_{m-1,im-1}, \{x_{mim}\}, \dots \rangle, i_m = \overline{1, n_m}$$

Путь на графе G из вершины X_1 к дуге x_{jij} j -го уровня будет $L = \langle x_{jij}, x_{j-1,ij-1}, \dots, x_{1/1} \rangle$. Каждой дуге ставится в соответствие значение некоторого условия. Множество решений i , значения j -го условия $R_{jij} = R(x_{jij})$. Тогда пути L_{jij} соответствует множество решений, которое существует для всех x_{rip} , $p = 1, j, i_p - 1, n_p$, т.е.

$$R(L_{jij}) = R(x_{2/2}) \cap R(x_{2/2}) \cap \dots \cap R(x_{rip}) \cap \dots \cap R(x_{jij}).$$

Для построения дуги, исходящей из i_{j-1} вершины j -го уровня, из таблицы соответствий выбирают условие X_j и для каждого значения этого условия x_{jij} проверяют соотношение: $R(x_{jij}) \cap R(L_{j-1,ij-1}) \neq \emptyset$. Если нет – дуга строится.

Последний уровень X_m содержит только висячие вершины (см. рис.2.19).

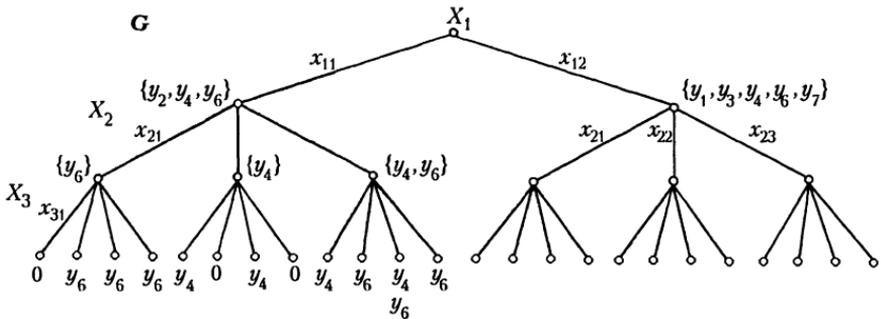


Рис.2.19. Граф-схема поиска решений

Число вершин равно:

$$1 + n_1 + n_1 \times n_2 + n_1 \times n_2 \times n_3 = 1 + \sum_{j=1}^m \prod_{j=1}^m n_j$$

Поиск решений по таблице соответствий

Исходной информацией является кортеж:

$$\bar{X} = \langle \bar{X}_1, \dots, \bar{X}_m \rangle$$

и таблица соответствий. Задача сводится к нахождению: $Y^* = f(\bar{X})$.

Нахождение решения сводится к поиску в таблице аргументов, в матрице соответствий и в таблице решений. Таблица аргументов имеет две строки, в верхней стоят названия условий, в нижней – значения условий. Рядом со значением условия, находится знак отношения: $\langle \Rightarrow \rangle$, $\langle \neq \rangle$, $\langle \Leftarrow \rangle$, $\langle \Rightarrow \rangle$, $\langle \leq \rangle$, $\langle \geq \rangle$.

Поиск в таблице аргументов

Задача: перейти от кортежа

к кортежу

$$a_p = \langle x_{1/1}, \dots, x_{jij}, \dots, x_{mim} \rangle.$$

При этом проверяется истинность высказывания

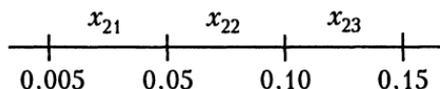
$$\bar{x}_j r_j x_{jij} \text{ для всех } i_j = \overline{1, n} \text{ и } j = \overline{1, m}; r_j - \text{знак отношения.}$$

Например, X_1 – вид поверхности $X_1 = \{0;1\}$, $r_1 = [=]$, $\bar{X}_1 = 1$ – кодирование по чертежу детали (1 – плоская поверхность, 0 – отверстие).

Проверяется $\bar{X}_1 = \left\{ \begin{array}{l} 1 = 0 - \text{ложно} \\ 1 = 1 - \text{истинно} \end{array} \right\}$, следовательно, X_2 – точ-

ность в пределах $0,005 \leq X_2 \leq 0,15$.

Диапазон точности разбивается по интервалам:



\bar{X}_2 задан в пределах $0,05 \leq \bar{X}_2 \leq 0,10 \Rightarrow \bar{X}_2 = x_{22}$.

Поиск в матрице соответствий сводится к нахождению результирующего столбца R , содержащего единичные клетки, которые присутствуют одновременно во всех столбцах с номерами

$$1i_1, 2i_2, \dots, j i_j, \dots, m i_m,$$

$$R = R_{1/1} \wedge R_{2/2} \wedge \dots \wedge R_{jij} \wedge \dots \wedge R_{mim}.$$

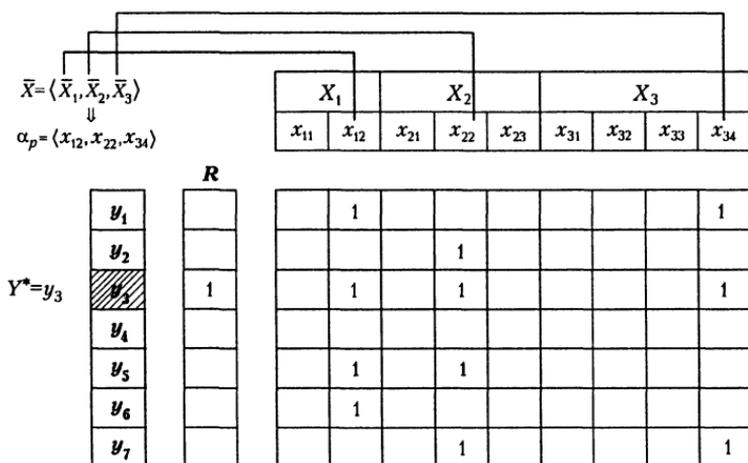
Поиск в таблице решений

Исходной информацией является столбец R и таблица решений – это столбец решений. Поиск заключается в определении номеров и названий решений.

$$Y \wedge T = y_l.$$

Схема поиска:

$$R = Rx_{12} \wedge Rx_{22} \wedge Rx_{34}.$$



2.2.4. Выбор варианта базирования, последовательности обработки поверхностей и построение размерной структуры ТП

Вначале дадим некоторые определения.

Определения

1. Путь на графе – последовательность ребер или дуг, в которых конец предыдущего совпадает с началом последующего.

2. Замкнутый путь, в котором конец последнего совпадает с началом первого. Называется контур для ориентированного графа или цикл для неориентированного.

3. Граф называется связным, если из любой вершины в любую другую существует путь.

4. Граф, не имеющий контуров или циклов, называется деревом.

5. Если ребро графа имеет направление, то такой граф называется ориентированным или орграфом.

6. Множество деревьев называется лесом.

7. Если в ориентированном дереве имеется вершина, к которой не подходит ни одна стрелка, то такое дерево называется корневым.

Каждой обрабатываемой поверхности ставится в соответствие поверхность, называемая технологической базой:

$$\text{нов. } jk \xrightarrow{f} \text{нов. } lr.$$

Установление соответствия f будем называть генерацией варианта базирования.

Выбор варианта базирования. Чтобы выбрать вариант базирования, надо иметь некоторое множество вариантов, из которого и осуществлять выбор. Выбор производят по некоторому критерию, т.е. каждому варианту ставится в соответствие некоторое число. Это позволяет упорядочить варианты и найти экстремальный. Например, критерием может быть число технологических баз (ТБ). Чем меньше число переустановок, тем вроде бы лучше. Или суммарный объем снимаемого припуска, который тоже желательно иметь меньше. В итоге желательно получить небольшое количество вариантов, которое может просмотреть и выбрать технолог как лицо, принимающее решение.

Вначале на примере рассмотрим какие графические структуры возникают при базировании и обработке детали (рис.2.20).

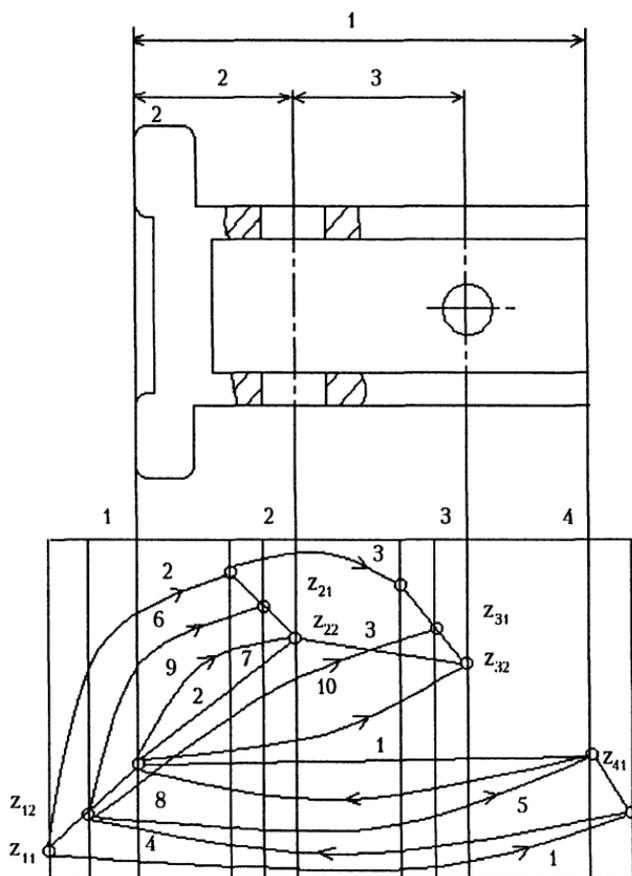


Рис.2.20. Деталь и граф ее механической обработки

$G_D = \langle 1,2,3 \rangle$ – остов чертежных размеров;

$G_z = \langle z_{11}, z_{12}, z_{21}, z_{22}, z_{31}, z_{32}, z_{41} \rangle$ – граф припусков, лес неориентированных деревьев;

$G_3 = \langle 1,2,3 \rangle$ – граф размеров заготовки, ориентированное корневое дерево;

$G_T = \langle 4,5,6,7,8,9,10 \rangle$ – граф технологических размеров, ориентированное корневое дерево;

$G = \langle G_D, G_z, G_3, G_T \rangle$ – граф размерной структуры технологического процесса.

G позволяет сделать следующие расчеты.

1. Найти все размерные цепи, замыкающимися звеньями которых являются размеры чертежа и припуски, составляющими звеньями служат размеры заготовки и технологические размеры.

Геометрическая интерпретация соотношений между номиналами и погрешностями (допусками) приведена на рис.2.21.

Для детали на рис.2.20 имеют место следующие соотношения:

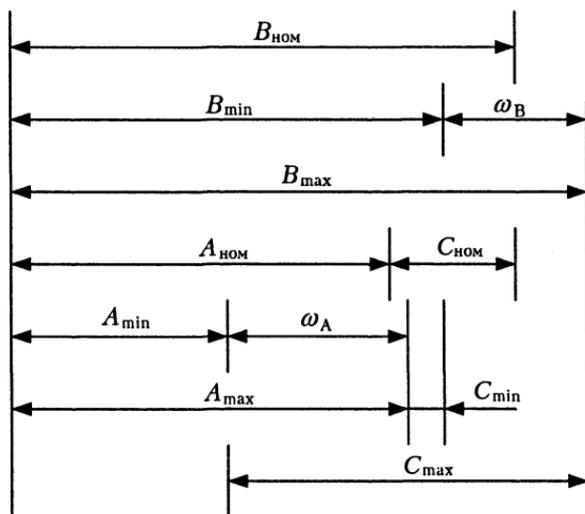


Рис.2.21. Соотношения номиналов и погрешностей в 3-х звенной размерной цепи

$$I = 8;$$

$$\omega_I = \omega_8$$

$$II = 9;$$

$$\omega_{II} = \omega_9$$

$$III = -9 + 10;$$

$$\omega_{III} = \omega_9 + \omega_{10}$$

$$Z_{11} = +1 - 4;$$

$$\omega_{Z11} = \omega_1 + \omega_4$$

$$Z_{12} = 5 + 8;$$

$$\omega_{Z12} = \omega_5 + \omega_8$$

$$\begin{aligned}
 Z_{21} &= -2 + 6 - 4 + 1; & \omega_{Z21} &= \omega_2 + \omega_6 + \omega_4 + \omega_1 \\
 Z_{22} &= -6 + 9 - 8 + 5; & \omega_{Z22} &= \omega_6 + \omega_9 + \omega_8 + \omega_5 \\
 Z_{31} &= -3 - 7 - 2 - 4 - 1; & \omega_{Z31} &= \omega_3 + \omega_7 + \omega_2 + \omega_4 + \omega_1 \\
 Z_{32} &= -7 + 10 - 8 + 5; & \omega_{Z32} &= \omega_7 + \omega_{10} + \omega_8 + \omega_5 \\
 Z_{41} &= +4 - 5; & \omega_{Z41} &= \omega_4 + \omega_5
 \end{aligned}$$

Погрешности размеров заготовки и технологических размеров, определяются применяемым технологическим оборудованием (моделью этого оборудования).

2. Рассчитать погрешности замыкающего звена – размера чертежа, которые служат для записи ограничений следующего вида:

$$\omega_I \leq T_I$$

$$T_{II} \leq T_{II} \quad T_i - \text{допуск на } i\text{-й размер детали.}$$

$$\omega_{III} \leq T_{III}$$

То есть план обработки и вариант базирования должны быть таковы, чтобы обрабатываемая деталь отвечала требованиям чертежа по точности.

3. Определить максимальные припуски, технологические размеры и размеры заготовки.

При помощи размерных цепей мы можем рассчитать все технологические размеры и все размеры заготовки.

Минимальный размер припуска (из технологических изображений):

$$Z_{MIN} = R_Z + T,$$

где R_Z – высота микронеровностей; T – толщина (глубина) дефектного слоя обрабатываемого металла.

Максимальный размер припуска:

$$Z_{MAX} = Z_{MIN} + \omega_Z.$$

ω_Z – погрешность припуска как замыкающего звена.

Пример:

Необходимо обработать заготовку (рис.2.22), чтобы получить размер I .

На рис.2.22 имеем:

$$Z_1 = 1 - 2$$

$$I = 2.$$

Дано: Чертеж детали, на котором заданы:

$$I_{MAX}; Z_{1MAX}$$

$$I_{MIN}; Z_{1MIN}$$

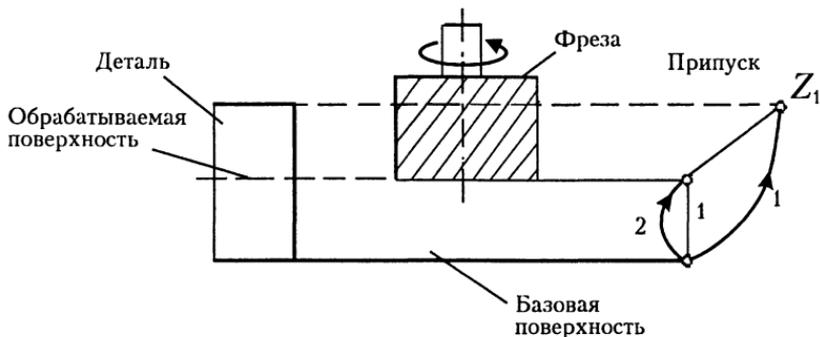


Рис.2.22. Обработка заготовки

Определить: 2_{MAX} ; 1_{MAX}

2_{MIN} ; 1_{MIN}

Количество неизвестных 4.

Составим уравнения:

$$1_{MAX} - 2_{MIN} = Z_{1MAX}$$

$$1_{MIN} - 2_{MAX} = Z_{1MIN}$$

$$2_{MAX} = I_{MAX}$$

$$2_{MIN} = I_{MIN}$$

Решая систему этих уравнений, найдем искомые неизвестные, т.е. технологические размеры и размеры заготовки.

В общем случае:

n_T – количество технологических размеров.

n_Z – количество размеров заготовки.

Отсюда:

$2(n_T + n_Z)$ – общее количество неизвестных;

$2(n_D + n_Z)$ – число уравнений,

где n_D – число размеров детали, n_Z – количество припусков.

Обозначим A_{Dj} – размер чертежа детали. Этот размер является замыкающим размером j -й размерной цепи. Размеры заготовки и технологические размеры могут принимать максимальные и минимальные значения.

Все звенья размерной цепи можно разбить на увеличивающие и уменьшающие.

Увеличивающее – звено, с увеличением которого замыкающее звено увеличивается.

Уменьшающее – звено, с увеличением которого замыкающее звено уменьшается.

Запишем систему уравнений для размерных цепей:

$$A_{Dj}^{\max} = \sum A_{Tzj}^{\rightarrow \max} - \sum A_{Tzj}^{\leftarrow \min}$$

$$A_{Dj}^{\min} = \sum A_{Tzj}^{\rightarrow \min} - \sum A_{Tzj}^{\leftarrow \max}$$

$$Z_i^{\max} = \sum A_{Tzi}^{\rightarrow \max} - \sum A_{Tzi}^{\leftarrow \min}$$

$$Z_i^{\min} = \sum A_{Tzi}^{\rightarrow \min} - \sum A_{Tzi}^{\leftarrow \max}.$$

Систему можно однозначно решить, если:

$$n_D + n_Z = n_T + n_3,$$

$$n_3 = n_D, \text{ если заготовка подобна детали.}$$

$$n_Z = n_T, \text{ если каждый припуск снимается своим технологическим размером.}$$

Генерация вариантов базирования осуществляется в следующей последовательности.

Строятся все возможные структуры, т.е. все графы:

$$G_i = \langle G_D, G_Z, G_3, G_{T_i} \rangle.$$

Графы должны обладать следующими свойствами.

1.1. Все поверхности, используемые как ТБ, должны обладать свойствами быть базой.

1.2 Граф G_{T_i} должен быть корневым деревом, покрывающим множество поверхностей (вершин), получаемых в процессе обработки.

1.3. При смене баз, определяемой деревом технологических размеров, каждая база в момент ее использования уже получена и еще не уничтожена (необходимое и достаточное условие).

2. Определяются допустимые размерные структуры.

2.1. Находятся все технологические размерные цепи с замыкающими звеньями – размерами детали.

2.2. Проверяется условие $\omega_{Adi} \leq T_{Adi}$ для всех размеров детали.

Генерация возможных структур

G_T – корневое дерево.

Будем предполагать, что поверхность заготовки используется в качестве ТБ только один раз. Количество корней – количество поверхностей заготовки, обладающих свойством быть базой.

Возникает задача: перечислить все корневые деревья, покрывающие заданное количество вершин. Этот алгоритм описан в книге Кристоридеса «Теория графов: Алгоритмический подход».

Рассмотрим генерацию на примере конкретного корневого дерева (рис.2.23, 2.24).

Мы должны покрыть пять вершин, не считая корневой, пятью размерами.

Рассмотрим более простой пример: четыре вершины и четыре размера. Крайние случаи: цепь и звезда. Перейдем от звезды к цепи (рис.2.25).

В каждом варианте число возможных последовательностей обхода поверхностей $P! = 1 \times 2 \times 3 \times 4 = 24$. Общее число вариантов $9 \times 24 = 216$.

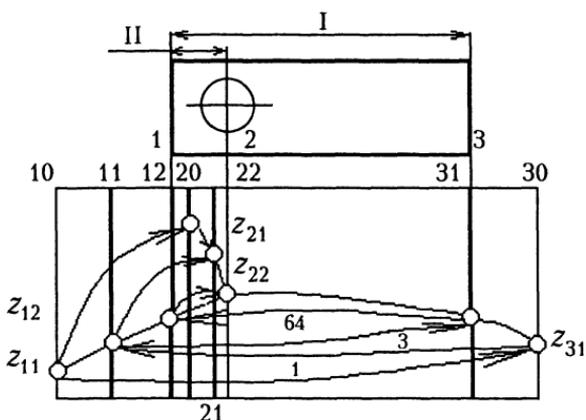


Рис.2.23. Деталь и граф ее механической обработки

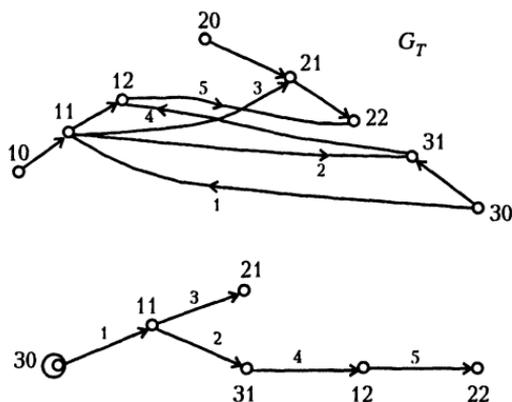


Рис.2.24. Граф размеров детали и технологических размеров

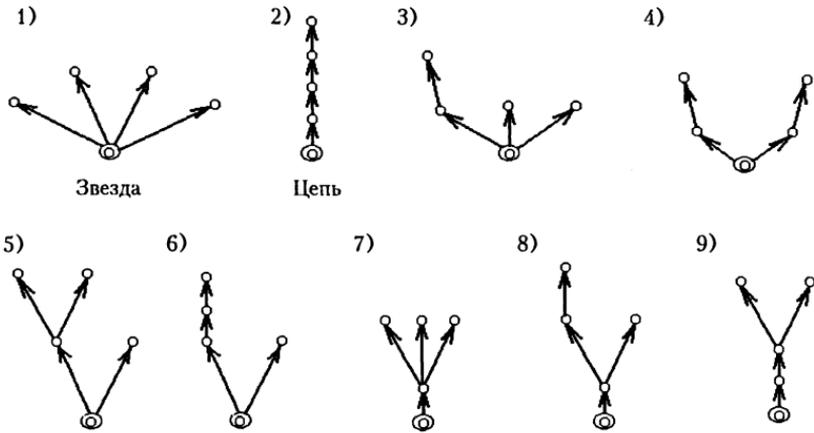


Рис.2.25. Различные варианты покрытий

Сгенерированные структуры должны отвечать, как уже говорилось, следующим условиям.

3. Промежуточные и окончательные поверхности, используемые в качестве технологических баз, должны удовлетворять свойству быть базой. Варианты, не удовлетворяющие этому условию, отбрасываются.

4. Варианты должны удовлетворять порядок следования припусков G_z , т.е. должны быть совместимы структуры G_z и G_{Ti} .

Требование к совместимости означает, что в структуре $G_z \cup G_{Ti}$ не должно быть замкнутых последовательностей поверхностей.

Дерево G_{Ti} может расти либо влево (и не расти вправо), либо вправо (и не расти влево) (рис.2.26).

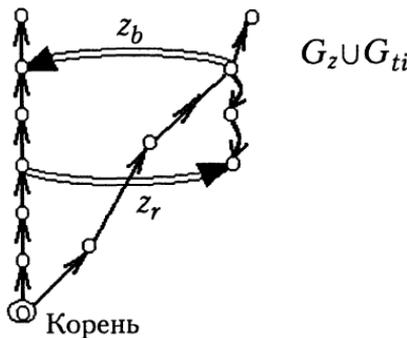


Рис.2.26. Графическое представление совместимости структур

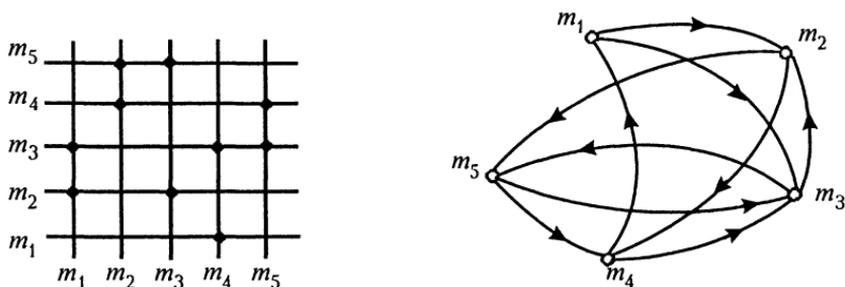


Рис.2.27. Представление графа на решетке и диаграммой

Использование критерия: количество технологических баз. Эта задача сводится к нахождению порядковой функции графа (рис.2.27).

Задание графа, по Бержу, делается при помощи соответствия Γ между элементами и подмножествами множества M .

$$\Gamma(m_1) = \{m_2, m_3\};$$

$$\Gamma(m_2) = \{m_4, m_5\};$$

$$\Gamma(m_3) = \{m_2, m_5\};$$

$$\Gamma(m_4) = \{m_1, m_3\};$$

$$\Gamma(m_5) = \{m_3, m_4\};$$

$$\Gamma(m_i) \rightarrow \text{образ } m_i.$$

Граф можно рассматривать как пару $G = \langle M, \Gamma \rangle$.

Граф G^{-1} определяется с помощью обратного отображения

$$\Gamma^{-1}; G^{-1} = \langle M, \Gamma^{-1} \rangle;$$

$$\Gamma^{-1}(m_1) = \{m_4\};$$

$$\Gamma^{-1}(m_2) = \{m_2, m_3\};$$

$$\Gamma^{-1}(m_3) = \{m_1, m_4, m_5\};$$

$$\Gamma^{-1}(m_4) = \{m_2, m_5\};$$

$$\Gamma^{-1}(m_5) = \{m_2, m_3\}.$$

Порядковая функция графа без контуров

Дан граф без контуров $G = \langle M, \Gamma \rangle$, определим подмножества $N_0, N_1, \dots, N_2, N_0 = \{X_i / X_i \subset M, \Gamma^{-1}(X_i) = \emptyset\}$ т.е. берутся те вершины из M , от которых стрелки только уходят

$N_1 = \{X_i / X_i \in M - N_0, \Gamma^{-1}(X_i) \subset N_0\}$ – начинаются в N_0 .

 $N_r = \{X_i / X_i \in M - N_0 \cup N_1 \cup \dots \cup N_{r-1}\}$

$\Gamma^{-1}(X_i) \subset N_0 \cup N_1 \cup \dots \cup N_{r-1}$,

где r – такое наименьшее число, что $\Gamma(N_r) = \emptyset$, т.е. N_r состоит из вершин, к которым стрелки только подходят.

Подмножества $N_k, k = 0, 1, \dots, r$ образуют разбиение M и вполне упорядочены соотношением $N_k < N_l, \Leftrightarrow K < L$.

Функция $O(X)$, определяемая равенством $X_i \in N_k \Rightarrow O(X_i) = K$, т.е. каждой вершине X , ставится в соответствие номер уровня K , называется порядковой функцией графа.

Рассмотрим построение порядковой функции графа на примере, используя алгоритм Демукрона: для графа механической обработки (рис.2.28) строим граф G_T и G_z .

Строим граф G_z и G_T .
 Образует булеву матрицу:

$$i, j = \begin{cases} 1, & \text{если } i \rightarrow j \\ 0, & \text{если нет.} \end{cases}$$

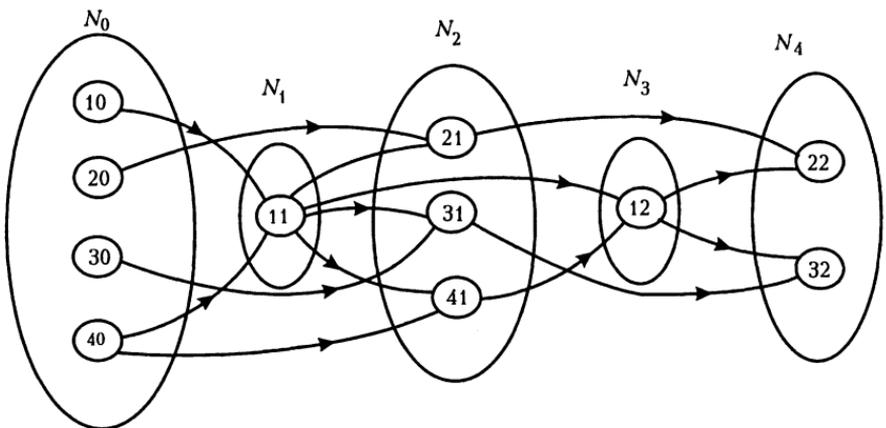


Рис.2.28. Разбиение графа G_T на уровни

| | 10 | 11 | 12 | 20 | 21 | 22 | 30 | 31 | 32 | 40 | 41 | |
|------------|----|----|----|----|----|----|----|----|----|----|----|---------|
| 10 | | 1 | | | | | | | | | | √ |
| 11 | | | 1 | | 1 | | | 1 | | | 1 | √ √ |
| 12 | | | | | 1 | | | | 1 | | | √ √ √ √ |
| 20 | | | | | 1 | | | | 1 | | | √ |
| 21 | | | | | | 1 | | | 1 | | | √ √ √ |
| 22 | | | | | | | | | | | | |
| 30 | | | | | | | | 1 | | | | √ |
| 31 | | | | | | | | | 1 | | | √ √ √ |
| 32 | | | | | | | | | | | | |
| 40 | | 1 | | | | | | | | 1 | | √ |
| 41 | | | 1 | | | | | | | 1 | | √ √ √ |
| Δ_0 | 0 | 0 | 2 | 0 | 2 | 2 | 0 | 2 | 2 | 0 | 2 | |
| Δ_1 | × | 2 | 2 | × | 1 | 2 | × | 1 | 2 | × | 1 | |
| Δ_2 | × | × | 1 | × | 0 | 2 | × | 0 | 2 | × | 2 | |
| Δ_3 | × | × | 0 | × | × | 1 | × | × | 1 | × | × | |
| Δ_4 | × | × | × | × | × | 0 | × | × | 0 | × | × | |

В строке Δ_0 нули – это вершины, из которых дуги уходят.

Уровень $N_0 = \{10, 20, 30, 40\}$

Уровень $N_1 = \{11\}$

Уровень $N_2 = \{21, 31, 41\}$

Уровень $N_3 = \{12\}$

Уровень $N_4 = \{22, 32\}$.

Порядковая функция позволяет подсчитать количество ТБ и порядок их смены.

Если граф содержит контур, то обязательно, появится строка без нулей. Поэтому мы строим граф $G_2 \cup G_T$, т.е. одновременно с порядковой функцией определяем совместимость структур G_2 и G_T .

2.2.5. Разложение графа на максимальные сильно связанные подграфы (приложение к задаче сборки или разузлования)

В некоторых задачах структурного анализа, т.е. где физическая (предметная) область моделируется графами, возникает проблема разбиения или разложения множества элементов системы на сильно связанные подмножества. Для решения этой задачи нам потребуются некоторые определения.

Транзитивное замыкание. Пусть задан граф $G = \langle M, \Gamma \rangle$ (по Бер-жу). Определим многозначные отображения по формулам:

$$\Gamma^2(m_i) = \Gamma\{\Gamma(m_i)\}$$

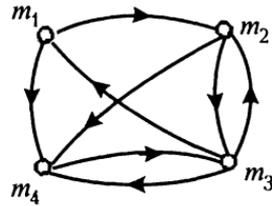
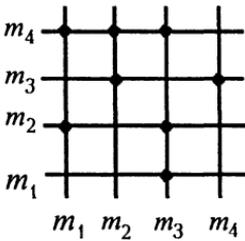
$$\Gamma^3(m_i) = \Gamma\{\Gamma^2(m_i)\} = \Gamma\{\Gamma\{\Gamma(m_i)\}\}$$

и обратные многозначные отображения по формулам:

$$\Gamma^{-2}(m_i) = \Gamma^{-1}\{\Gamma^{-1}(m_i)\}$$

$$\Gamma^{-3}(m_i) = \Gamma^{-1}\{\Gamma^{-2}(m_i)\} = \Gamma^{-1}\{\Gamma^{-1}\{\Gamma^{-1}(m_i)\}\}.$$

Пример:



$$\Gamma(m_1) = \{m_2, m_4\}, \Gamma^2(m_1) = \Gamma(m_2) \cup \Gamma(m_4) = \{m_3, m_4\}$$

$$\Gamma^3(m_1) = \Gamma(m_3) \cup \Gamma(m_4) = \{m_1, m_2, m_3, m_4\}$$

$$\Gamma^{-1}(m_1) = \{m_3\}; \Gamma^{-2}(m_1) = \Gamma^{-1}(m_3) = \{m_2, m_4\}$$

$$\Gamma^{-3}(m_1) = \{m_1, m_2, m_3\}.$$

Очевидно, что $\Gamma^n(m_i)$ и $\Gamma^{-n}(m_i)$ являются подмножествами тех вершин, в которые можно прийти из m_i , используя пути длины n (или меньше), и тех вершин, из которых можно прийти в m_i , используя пути длины n (или меньше).

Транзитивным замыканием $\Gamma(m_i)$ называется многозначное отображение, определяемое формулой

$$\hat{\Gamma}(m_i) = \{m_i\} \cup \Gamma(m_i) \cup \Gamma^2(m_i) \cup \Gamma^3(m_i) \cup \dots,$$

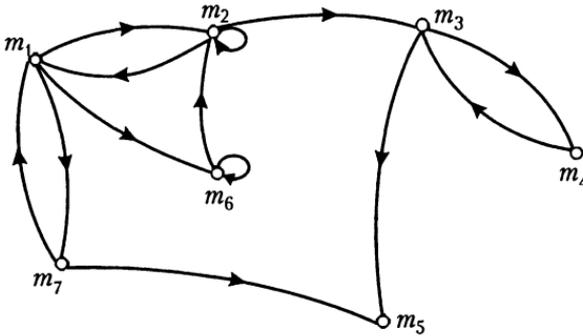
т.е. $\hat{\Gamma}(m_i)$ – множество вершин, в которое можно прийти из m_i по некоторому пути.

Аналогично определяется обратное транзитивное замыкание:

$$\hat{\Gamma}^{-1}(m_i) = \{m_i\} \cup \Gamma^{-1}(m_i) \cup \Gamma^{-2}(m_i) \cup \Gamma^{-3}(m_i) \cup \dots$$

т.е. множество вершин, из которых попадают в m_i по некоторому пути.

Пример:



для приведенного выше графа.

$$\Gamma(m_1) = \{m_2, m_6, m_7\}; \Gamma^2(m_1) = \{m_1, m_2, m_3, m_6, m_5\}$$

$$\Gamma^3(m_1) = \{m_2, m_6, m_7, m_1, m_3, m_4, m_5\} = M = \hat{\Gamma}(m_1)$$

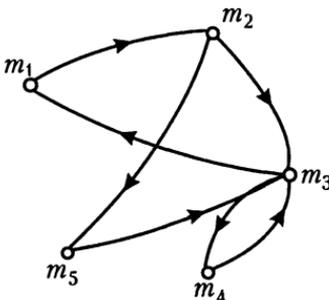
$$\Gamma^{-1}(m_1) = \{m_7\}; \Gamma^{-2}(m_1) = \{m_1, m_2, m_6\}$$

$$\Gamma^{-3}(m_1) = \{m_2, m_7, m_1, m_6\}$$

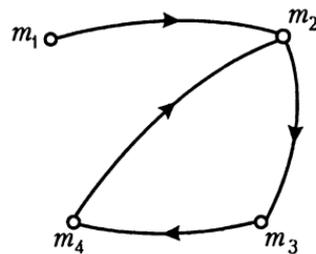
$$\Gamma^{-4}(m_1) = \{m_2, m_1, m_6, m_7\} = \hat{\Gamma}^{-1}(m_1).$$

Сильно связный граф. Граф $G = \langle M, \Gamma \rangle$ называют сильно связным, если $(\forall m_i \in M) \hat{\Gamma}(m_i) = M$, т.е. для любых двух вершин m_i, m_j такого графа существует путь, идущий из m_i в m_j .

Пример сильно связных и несильно связных графов:



Сильно связный граф



Несильно связный граф

Приведенное условие эквивалентно условию:

$$(\forall m_i \in M) \hat{\Gamma}^{-1}(m_i) = M.$$

Разложение графа на максимальные сильно связанные подграфы

Подграфом G' графа $G = \langle M, \Gamma \rangle$ называется граф $G' = \langle M', \Gamma_2 \rangle$, если $M' \subset M$ и $(\forall m_i \in M') \Gamma_2(m_i) = M' \cap \Gamma(m_i)$.

Подграф G' строится на подмножестве вершин графа G и включает все дуги, входящие и выходящие из этого подмножества.

Подграф называют *максимальным сильно связным*, если не существует сильно связного графа G'' , строго содержащего G' .

Метод разложения графа на максимальные сильно связанные подграфы

Если $C(m_i)$ – подграф, содержащий m_i , то

$$C(m_i) = \hat{\Gamma}(m_i) \cap \Gamma^{-1}(m_i).$$

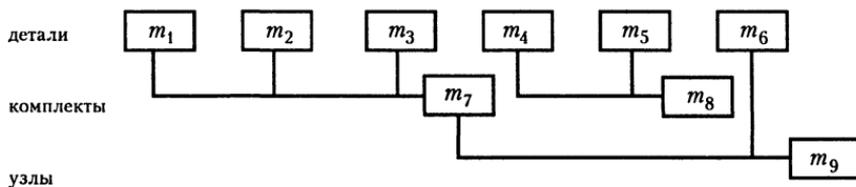
Метод состоит в следующем:

Берем произвольную вершину m_i и находим $\hat{\Gamma}(m_i)$, $\hat{\Gamma}^{-1}(m_i)$ и $\hat{\Gamma}(m_i) \cap \hat{\Gamma}^{-1}(m_i)$.

Затем берем вершину $m_j \notin C(m_i)$ и действуем аналогично.

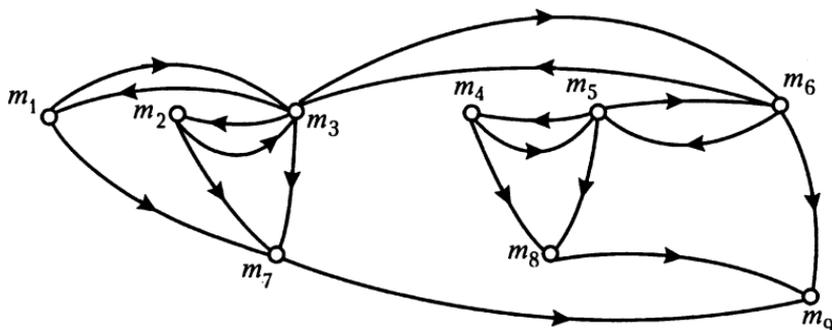
Процесс продолжается до тех пор, пока это возможно.

Рассмотрим алгоритм на примере схемы сборки:



В этом примере m_3, m_5, m_6 – базовые детали.

Преобразуем схему сборки в граф.



Строим матрицу смежностей.

| | m_1 | m_2 | m_3 | m_4 | m_5 | m_6 | m_7 | m_8 | m_9 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| m_1 | | | 1 | | | | 1 | | |
| m_2 | | | 1 | | | | 1 | | |
| m_3 | 1 | 1 | | | | 1 | 1 | | |
| m_4 | | | | | 1 | | | 1 | |
| m_5 | | | | 1 | | | | 1 | |
| m_6 | | | 1 | | | | | | 1 |
| m_7 | | | | | | | | | 1 |
| m_8 | | | | | | | | | 1 |
| m_9 | | | | | | | | | |

Длины путей

| $\hat{\Gamma}(m_1)$ |
|---------------------|
| 0 |
| 2 |
| 1 |
| 4 |
| 3 |
| 2 |
| 1 |
| 4 |
| 2 |

Длины путей

| $\hat{\Gamma}^{-1}(m_1)$ | 0 | 2 | 1 | 4 | 3 | 2 | x | x | x |
|--------------------------|---|---|---|---|---|---|---|---|---|
|--------------------------|---|---|---|---|---|---|---|---|---|

$$\hat{\Gamma}^{-1}(m_1) = \{m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, m_9\}$$

$$\hat{\Gamma}^{-1}(m_1) = \{m_1, m_2, m_3, m_4, m_5, m_6\}$$

$$C(m_1) = \{m_1, m_2, m_3, m_4, m_5, m_6\}$$

Удаляем из графа вершины $C(m_1)$.

| | m_7 | m_8 | m_9 |
|-------|-------|-------|-------|
| m_7 | | | 1 |
| m_8 | | | 1 |
| m_9 | | | |

| $\hat{\Gamma}(m_7)$ |
|---------------------|
| 0 |
| x |
| 1 |

$\hat{\Gamma}(m_7) = \{m_7, m_9\}$,

$\hat{\Gamma}^{-1}(m_7) = \{m_7\}$,

$C_3(m_7) = \{m_7\}$.

| $\hat{\Gamma}^{-1}(m_7)$ | 0 | x | x |
|--------------------------|---|---|---|
|--------------------------|---|---|---|

Удаляем из графа вершины $C(m_7)$.

| | m_8 | m_9 |
|-------|-------|-------|
| m_8 | | 1 |
| m_9 | | |

| $\hat{\Gamma}(m_8)$ |
|---------------------|
| 0 |
| 1 |

$\hat{\Gamma}(m_8) = \{m_8, m_9\}$,

$\hat{\Gamma}^{-1}(m_8) = \{m_8\}$,

$C_3(m_8) = \{m_8\}$.

| $\hat{\Gamma}^{-1}(m_8)$ | 0 | x |
|--------------------------|---|---|
|--------------------------|---|---|

Возникает четыре класса или четыре максимальные сильно связанные подграфы:

$$C(m_1) = \{m_1, m_2, m_3, m_4, m_5, m_6\}$$

$$C(m_7) = \{m_7\}$$

$$C(m_8) = \{m_8\}$$

$$C(m_9) = \{m_9\}.$$

Таким образом, имея конструкторскую информацию о базировании деталей в конструкции, мы можем получить технологическую информацию о схеме сборки.

2.2.6. Конечные автоматы (приложение к работе сборочного конвейера)

Пусть $X = \{x_1, x_2, \dots, x_m\}$ и $Y = \{y_1, y_2, \dots, y_n\}$ – два произвольных множества элементов, которые будем называть алфавитами, а их элементы – буквами алфавита.

Конечную упорядоченную последовательность букв, будем называть *словом* в данном алфавите. $\sigma(X)$ и $\sigma(Y)$ соответственно, множества слов в алфавитах X и Y . Тогда произвольное преобразование дискретной информации можно задавать как однозначное отображение f множества слов $\sigma(X)$ в множество слов $\sigma(Y)$. Отображение f обычно называют алфавитным отображением, или алфавитным оператором, а алфавиты X и Y – соответственно входным и выходным алфавитами оператора f .

Каждому входному слову оператор f ставит соответствие выходное слово $r = y_{j_1}, y_{j_2}, \dots, y_{j_i}$. Поэтому для каждого $P \in \sigma(X)$ существует $r \in \sigma(Y)$, такое что $r = f(P)$. В этом случае f есть функция, область определения которой $\sigma(X)$, а область значений $\sigma(Y)$.

Любой абстрактный автомат реализует некоторый оператор f или, как говорят, индуцирует некоторое отображение f .

Отметим ряд допущений, связанных с понятием абстрактного автомата.

1. Наличие произвольного числа отличных друг от друга состояний автомата и свойства мгновенного перехода из одного состояния в другое.

2. Переход из одного состояния в другое оказывается возможным не ранее, чем через некоторый промежуток времени $\Delta (\Delta > 0)$ – интервал дискретности.

3. Число различных входных и выходных сигналов конечно.

4. Входные сигналы – причина перехода автомата из одного состояния в другое, а выходные сигналы – реакция автомата на

входные сигналы – относятся к моментам времени, определяемым соответствующими переходами автомата.

Учитывая эти обстоятельства, можно говорить, что абстрактный автомат функционирует в дискретном времени $t = 0, 1, 2, \dots$. На каждый входной сигнал $x(t)$ $t > 0$, автомат реагирует сигналом $y(t)$.

Различают два вида автоматов – синхронные и асинхронные.

В синхронных автоматах переход из одних состояний в другие осуществляется через равные промежутки времени, в то время как в асинхронных автоматах эти переходы совершаются через произвольные промежутки времени.

Остановимся более подробно на законах функционирования автоматов. Состояние $q(t)$ автомата в момент времени t однозначно определяется предыдущим состоянием $q(t - 1)$ и входным сигналом $x(t)$. Поэтому можно записать:

$$q(t) = \varphi(q(t-1), x(t)),$$

где φ – функция, определяющая последующее состояние автомата, которая обозначается просто $\varphi(q, x)$ и называется *функцией переходов*.

Выходной сигнал $y(t)$ реального автомата всегда появляется после входного сигнала $x(t)$. Однако, относительно времени перехода t из состояния $q(t - 1)$ в состояние $q(t)$ выходной сигнал $y(t)$ может появиться либо раньше, либо позже этого времени. Поэтому справедливы выражения:

$$y(t) = \psi(q(t - 1), x(t)),$$

$$y(t) = \psi(q(t), x(t)),$$

где $\psi(q, x)$ – функция выходов обычная ($q(t - 1)$) или сдвинутая ($q(t)$).

Различают:

автомат первого рода

$$q(t) = \varphi(q(t - 1), x(t))$$

$$y(t) = \psi(q(t - 1), x(t))$$

автомат второго рода

$$q(t) = \varphi(q(t - 1), x(t))$$

$$y(t) = \psi(q(t), x(t)), \text{ где } t = 1, 2, 3, \dots$$

Способы задания автоматов. Существует несколько эквивалентных способов задания автоматов, по аналогии с графами их три: аналитический, геометрический, матричный.

Говорят, что задан абстрактный автомат, если задано:

1. $X = \{x_i\}$, $i = \overline{1, m}$ – входной алфавит;
2. $Y = \{y_j\}$, $j = \overline{1, n}$ – выходной алфавит;
3. $Q = \{q_k\}$, $k = \overline{1, p}$ – алфавит состояний;
4. $F: X \times Q \rightarrow Q$ – операция, переводящая декартово произведение $F \times Q$ в Q , или $F \div X \times Q \rightarrow \langle Q, Y \rangle$;
5. $q_1 \in Q$ – начальное состояние автомата.

Другое задание:

1. $X = \{x_i\}$, $i = \overline{1, m}$
2. $Y = \{y_i\}$, $j = \overline{1, n}$
3. $Q = \{q_k\}$, $k = \overline{1, p}$
4. $q = f(x, q)$ – функция переходов.
5. $y = \psi(x, q)$ – функция выходов.

Перечисленные способы задания являются аналитическими. Они состоят из трех конечных множеств и двух функций, определенных на множествах.

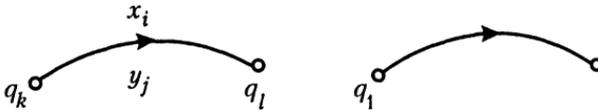
Множества X , Y , Q являются не только конечными, но и счетными, эквивалентными натуральному ряду. Поэтому матричное задание автомата состоит в табличном задании функций $q = f(x, q)$, $y = \psi(x, q)$.

| $x \setminus q$ | q_1 | q_2 | ... | q_k | q_p |
|-----------------|-------|-------|-----|-------|-------|
| x_1 | | | | | |
| x_2 | | | | | |
| \vdots | | | | | |
| x_i | | | | q_i | |
| \vdots | | | | | |
| x_m | | | | | |

| $x \setminus q$ | q_1 | q_2 | ... | q_k | q_p |
|-----------------|-------|-------|-----|-------|-------|
| x_1 | | | | | |
| x_2 | | | | | |
| \vdots | | | | | |
| x_i | | | | y_i | |
| \vdots | | | | | |
| x_m | | | | | |

Геометрический способ задания автомата сводится к изображению ориентированного графа, вершинами которого являются состояния автомата, обозначаемые различными буквами: $q \in Q$, а около каждого ребра $\langle q_k, q_j \rangle$ ставится буква входного алфавита $x_i \in X$, вызывающая переход автомата из состояния q_k в q_j , и буква выходного алфавита $y_i \in Y$, которая появляется на выходе автомата. Если

автомат первого рода, то выходная буква определяется парой (q_k, x_i) , а если второго рода, то парой (q_l, x_i) . Начальное состояние автомата обозначается буквой $q_1 \in Q$.



Таким образом, ориентированный граф, ребра которого нагружены буквами входного и выходного алфавитов, однозначно задает некоторый абстрактный автомат. Графы с нагруженными ребрами (весовыми функциями на ребрах) обычно называют графоидами. Поэтому ориентированный графоид – геометрическая интерпретация абстрактного автомата.

Пример способов задания абстрактного автомата.

Пусть задан абстрактный автомат (аналитическое задание):

$$A = \langle X, Y, Q, q = f(x, q), y = \psi(x, q) \rangle,$$

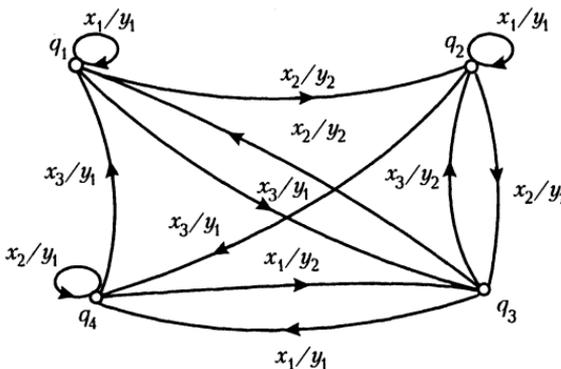
$$X = \{x_1, x_2, x_3\}, Y = \{y_1, y_2\}, Q = \{q_1, q_2, q_3, q_4\}$$

Функции f и ψ удобнее задавать таблично, отсюда табличный способ задания автомата:

| $x \setminus q$ | q_1 | q_2 | q_3 | q_4 |
|-----------------|-------|-------|-------|-------|
| x_1 | q_1 | q_2 | q_4 | q_3 |
| x_2 | q_2 | q_3 | q_1 | q_4 |
| x_3 | q_3 | q_4 | q_2 | q_1 |

| $x \setminus q$ | q_1 | q_2 | q_3 | q_4 |
|-----------------|-------|-------|-------|-------|
| x_1 | y_1 | y_2 | y_1 | y_2 |
| x_2 | y_2 | y_1 | y_2 | y_1 |
| x_3 | y_1 | y_1 | y_2 | y_1 |

Геометрический способ задания автомата:



Предположим, что автомат является автоматом первого рода. Подача на вход автомата A , установленного в начальное состояние q_1 входного слова $P_1 = x_1 x_1 x_2 x_3 x_2 x_3$ вызывает на выходе слово $r_1 = y_1 y_1 y_2 y_2 y_2 y_1$.

Пример описания реального автомата абстрактным.

Из цехов A и B идут детали A и B в некоторой беспорядочной последовательности $AABABBAABA...$. Требуется образовать тройки для последующей сборки (рис.2.29).

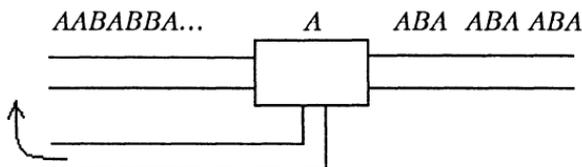


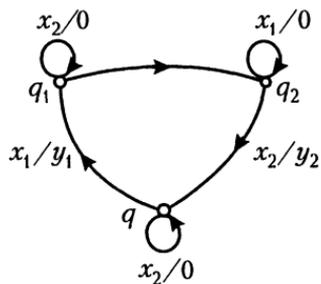
Рис.2.29. Схема сборочного автомата

$X = \{x_1 = A, x_2 = B\}$; $Y = \{y_1 = A, y_2 = B, y_3 = 0\}$; $Q = \{q_1, q_2, q_3\}$.

q_1 — автомат ждет деталь $A = x_1$;

q_2 — автомат A ждет $B = x_2$;

q_3 — автомат после AB ждет $A = x_1$.



Графoid автомата

| $q \backslash x \ y$ | x_1 | x_2 | y_1 | y_2 |
|----------------------|-------|-------|-------|-------|
| q_1 | q_2 | q_1 | y_1 | 0 |
| q_2 | q_2 | q_3 | 0 | y_2 |
| q_3 | q_1 | q_3 | y_1 | 0 |

Матрица переходов и выходов

2.2.7. Непрерывные модели

Представление технологической системы (ТС) в виде ориентированного объекта. Рассмотрим, от чего зависит погрешность каждого составляющего звена технологической размерной цепи. Для этого представим ТС в виде ориентированного объекта (рис.2.30).

\bar{x}_{ji}^{BX} — входное положение элемента формы заготовки, относительно технологической базы, свойства материала заготовки.

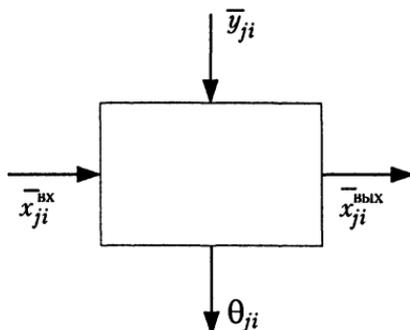


Рис.2.30. Технологическая система

$\bar{x}_{ji}^{вых}$ – выходное положение элемента формы заготовки, относительно технологической базы, переменные ТП.

\bar{y}_{ji} – управляющие переменные ТП.

θ_{ji} – критерий оптимальности.

Пример: токарная обработка (рис.2.31).

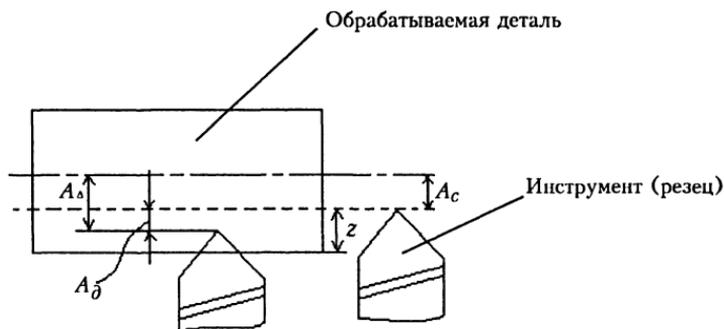


Рис.2.31. Схема токарной обработки

Здесь:

A_c – размер статической настройки, A_{δ} – размер динамической настройки, A_{Δ} – технологический размер, $A_{\Delta} = A_c + A_{\delta}$.

В рассматриваемом примере переменные, описывающие ТП, следующие:

$$x^{вых} = \langle A_{\delta}, R_a, h_{зад} \rangle; x^{вх} = \langle z, H \rangle; Y = \langle V, S \rangle,$$

где R_a – шероховатость поверхности детали, $h_{зад}$ – износ инструмента, z – припуск на обработку, H – твердость материала заготовки, V – скорость резания, S – подача.

Отдельного рассмотрения заслуживает критерий θ . Одним из возможных критериев является себестоимость работы ТС по преобразованию заготовки в готовую деталь. Она складывается из двух частей:

$$\theta = E_c t_M + E_{II} \frac{t_M}{T},$$

где E_c – стоимость станкоминуты, E_{II} – стоимость работы инструмента за T , t_M – машинное время, T – период стойкости инструмента.

$$t_M = \frac{l}{S_0 \cdot n} = \frac{l}{S_M} = \frac{\pi dl}{1000VS};$$

$$n = \frac{1000V}{\pi d}; \quad \theta = \frac{\pi dl}{1000VS} E_c + E_{II} \frac{\pi dl}{1000VST} = \frac{C_0}{VS} \text{ при } T = \text{const};$$

$$C_0 = \frac{\pi dl}{1000} \left(E_c + \frac{E_{II}}{T} \right),$$

где S_0 – подача на оборот детали, S_M – минутная подача.

Таким образом, можно описать ТС следующими зависимостями:

$$\begin{aligned} x^{\text{вых}} = F(x^{\text{вх}}, Y) \quad \text{или} \quad A_d = f_1(V, S, z, H), \\ R_a = f_2(V, S, z, H), \\ h_{\text{зад}} = f_3(V, S, z, H), \\ \theta = \Phi(x^{\text{вх}}, Y) \quad \text{или} \quad \theta = \varphi(V, S). \end{aligned}$$

Подобные модели в отличие от графовых называются непрерывными, или параметрическими.

Задачи статистического моделирования (применительно к моделям ТС). Под моделированием или выяснением механизма явлений мы будем понимать следующую задачу. Имеется некоторый объект A , обладающий входом x и выходом y . Объект A переводит множество входных величин X в множество выходных Y . Элементы множеств X и Y могут быть скалярами, векторами, функциями. Требуется определить структуру объекта A на всем множестве X , или вид преобразования.

Возможно два различных подхода: аналитический и статистический. ТП – есть пример сложной диффузной системы. Будем рассматривать вход – вектор, выход – скаляр, т.е. $y = f(x_1, x_2, \dots, x_n)$. При этом математическое описание – отрезок ряда Тейлора.

$$y = b_0 + \sum_{i=1}^n b_i x_i + \sum_{i<j}^n b_{ij} x_i x_j + \sum_{i=1}^n b_{ii} x_i^2 + \dots,$$

$$\text{где } b_0 = y_0; b_i = \frac{\partial f}{\partial x_i}; b_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}; b_{ii} = \frac{\partial^2 f}{\partial x_i^2}.$$

Вводя новые переменные типа: $x_{n+1} \equiv x_1 x_2$; $x_{n+2} = x_1 x_3$; $x_{n+r} = x_1^2$ и т.п., приходим к искусственной линейной форме вида:

$$y = b_0 x_0 + b_1 x_1 + b_2 x_2 + \dots + b_k x_k, \text{ где } x_0 \equiv 1.$$

Задача: определить коэффициенты b_0, b_1, \dots, b_k по результатам эксперимента.

$$(x_{11} x_{12} \dots x_{1k}) \rightarrow y_1; y_1 = b_0 x_0 + b_1 x_{11} + \dots + b_k x_{1k} + e_1$$

$$(x_{21} x_{22} \dots x_{2k}) \rightarrow y_2; y_2 = b_0 x_0 + b_1 x_{21} + \dots + b_k x_{2k} + e_2$$

$$(x_{N1} x_{N2} \dots x_{Nk}) \rightarrow y_N; y_N = b_0 x_0 + b_1 x_{N1} + \dots + b_k x_{Nk} + e_N.$$

Вводим матричные обозначения:

$$\begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_N \end{pmatrix} = Y; \begin{pmatrix} b_0 \\ b_1 \\ \dots \\ b_k \end{pmatrix} = B; \begin{pmatrix} e_0 \\ e_1 \\ \dots \\ e_N \end{pmatrix} = E; \begin{pmatrix} x_{10} & x_{11} & x_{1k} \\ x_{20} & x_{21} & x_{2k} \\ \dots & \dots & \dots \\ x_{N0} & x_{N1} & x_{Nk} \end{pmatrix} = X,$$

X – матрица плана эксперимента, x_{ij} – точки факторного пространства (точки пространства эксперимента).

$$\text{Тогда } Y = XB + E.$$

Вспомним матричные операции.

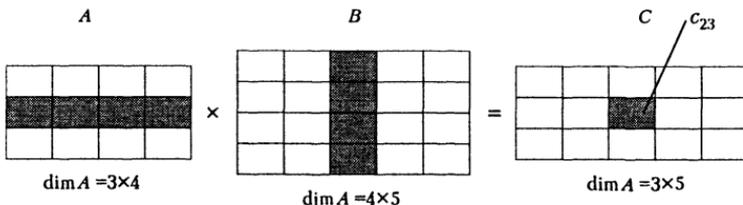
Произведение

$A \times B = C \Rightarrow C_{ik} = A_i B_k$ – скалярное произведение строки на столбец. Количество столбцов матрицы A равно количеству строк матрицы B .

$$\dim A = p \times q, \dim B = q \times s, \dim C = p \times s.$$

В скалярной форме

$$c_{ik} = \sum_{j=1}^q a_{ij} b_{ji}$$



Произведение матриц некоммутативно. Скалярное произведение векторов коммутативно:

$$a_1 b_1 = a_2 b_2 = b_1 a_1 + b_2 a_2.$$

Транспонирование – замена строк столбцами, и наоборот.

A и A' . Свойство $(AB)' = B' A'$.

Доказательство:

$$A \times B = C, C_{ik} = A_k B_k, \text{ стр. столб.}$$

$$C'_{ik} = C_{ki} = A_k B_i = B_i A_k = B'_i A'_k. \text{ стр. столб. столб. стр. стр. столб.}$$

A^{-1} . Свойство обратной матрицы. Обратная матрица $AA^{-1} = A^{-1}A = 1$ – единичная матрица.

Скалярное произведение

$$E' E = (e_1 e_2 \dots e_N) \times \begin{pmatrix} e_1 \\ e_2 \\ \dots \\ e_N \end{pmatrix} = e_1^2 + e_2^2 + \dots + e_N^2$$

– скалярный квадрат вектора. Выражение для скалярного квадрата будет

$$Y = XB + E \Rightarrow E = Y - XB; E' = Y' - B'X';$$

$$\begin{aligned} E'E &= (Y' - B'X')(Y - XB) = Y'Y - B'X'Y - Y'XB + B'X'XB = \\ &= Y'Y - 2B'X'Y + B'X'XB = S_R \end{aligned}$$

– остаточная сумма квадратов.

Здесь мы использовали свойство $B'X'Y = Y'XB$, так как $A'B = B'A$.

Таким образом, $S_R = f(B)$. Чтобы найти \min , надо

$$\frac{\partial S_R}{\partial B} = 0 = \begin{pmatrix} \frac{\partial S_R}{\partial b_0} \\ \frac{\partial S_R}{\partial b_1} \\ \dots \\ \frac{\partial S_R}{\partial b_n} \end{pmatrix} = 0.$$

То есть нам надо продифференцировать выражение для S_R .

Вспомним элементы векторного дифференцирования.

$$\frac{\partial Y' Y}{\partial B} = 0; \quad \frac{\partial 2B' X' Y}{\partial B} = 2X' Y; \quad X' Y = A = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}; \quad B' = (b_0 b_1);$$

$$B' A = (b_0 b_1) \times \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = b_0 a_1 + b_1 a_2; \quad \frac{\partial B' A}{\partial B} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix};$$

$$\frac{\partial B' X' X B}{\partial B} = 2X' X B.$$

$X' X$ – симметрическая матрица, так как $(X' X)' = X' X$;

$$X' Y = A = \begin{pmatrix} a_{11} & a_{21} \\ a_{21} & a_{22} \end{pmatrix}; \quad \text{Причем } a_{12} = a_{21} \quad B = \begin{pmatrix} b_0 \\ b_1 \end{pmatrix};$$

$$\begin{aligned} (b_0 \quad b_1) \times \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \times \begin{pmatrix} b_0 \\ b_1 \end{pmatrix} &= (b_0 \quad b_1) \times \begin{pmatrix} a_{11}b_0 + a_{12}b_1 \\ a_{21}b_0 + a_{22}b_1 \end{pmatrix} = \\ &= a_{11}b_0^2 + a_{12}b_0b_1 + a_{21}b_0b_1 + a_{22}b_1^2 \end{aligned}$$

– квадратичная форма вектора B .

Дифференцируем по b_0 и b_1 .

$$\begin{pmatrix} a_{11}b_0 + a_{12}b_1 \\ a_{21}b_0 + a_{22}b_1 \end{pmatrix} = 2 \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix} \times \begin{pmatrix} b_0 \\ b_1 \end{pmatrix} = 2AB.$$

$$\text{Отсюда } \frac{\partial S_R}{\partial B} = -2X' Y + 2X' X B \Rightarrow X' X B = X' Y;$$

$$(X' X)^{-1} (X' X) B = (X' X)^{-1} X' Y B = (X' X)^{-1} X' Y.$$

Проведенные преобразования носят название метода наименьших квадратов – М.Н.К.

Свойства оценок, полученных М.Н.К.

1. B – являются линейными оценками.
2. B – являются несмещенными оценками.

Доказательство: $M[Y] = X\beta$;

$$\begin{aligned} M[B] &= M[(X' X)^{-1} X' Y] = (X' X)^{-1} X' M[Y] = \\ &= (X' X)^{-1} X' X \beta = \beta; \quad M[E] = 0. \end{aligned}$$

3. В классе линейных несмещенных оценок оценки, полученные методом М.Н.К., обладают минимальной дисперсией (теорема Гаусса–Маркова).

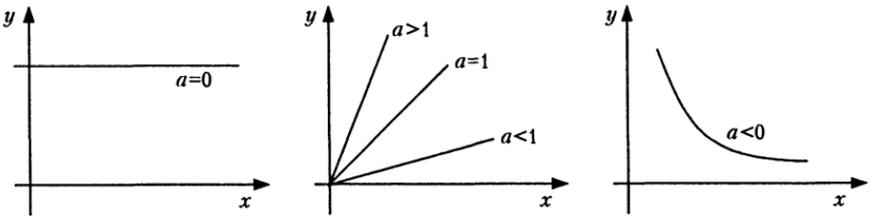
Планирование экспериментов. М.Н.К. минимизирует дисперсию на множестве методов вычисления оценок. Дисперсия также зависит от расположения точек в факторном пространстве. Минимум дисперсии доставляет ортогональный план, задаваемый ортогональной матрицей (теорема Бокса).

Рассмотрим на примере получение ортогонального плана и расчет коэффициентов B .

Конкретизируем вид зависимости $Ag = f(V, S, Z)$.

Выдвинем гипотезу, что $Ag = \beta_0 V^{\beta_1} S^{\beta_1} Z^{\beta_1}$.

Степенная зависимость выдвинута из следующих соображений для степенной функции $y = cx^a$ при различных a получаем следующие различные графики:



Переменные меняются в некоторой области, задаваемой неравенствами:

$$\left. \begin{array}{l} V_{\min} \leq V \leq V_{\max} \\ S_{\min} \leq S \leq S_{\max} \\ Z_{\min} \leq Z \leq Z_{\max} \end{array} \right\}$$

параллелепипед в пространстве факторов V, S, Z .

Логарифмированием выражение для Ag приводится к линейному относительно логарифмов.

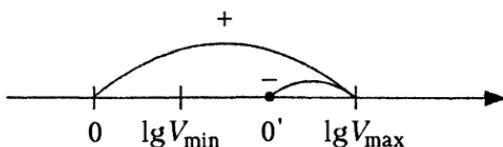
Введем обозначения:

$$x_1 = \frac{2 \lg V - \lg V_{\max}}{\lg V_{\max} - \lg V_{\min}} + 1;$$

$$x_2 = \frac{2 \lg S - \lg S_{\max}}{\lg S_{\max} - \lg S_{\min}} + 1;$$

$$x_3 = \frac{2 \lg Z - \lg Z_{\max}}{\lg Z_{\max} - \lg Z_{\min}} + 1.$$

Принцип получения формул поясняется графиком:



и последовательностью действий.

1. Перенос начала координат в точку $0'$:

$x'_1 = \lg V - 0'$, посередине между min и max;

$$0' = \lg V_{\max} - \frac{\lg V_{\max} - \lg V_{\min}}{2} =$$

$$= \lg V_{\max} - \frac{\Delta V}{2} \Rightarrow x'_1 = \lg V - \lg V_{\max} + \frac{\Delta V}{2}.$$

2. Сжатие шкалы, т.е. преобразование подобия, $x_1 = Kx'_1$, причем K находим из условия $x_1 = 1$ при $\lg V = \lg V_{\max}$.

$$1 = K \left(\lg V_{\max} - \lg V_{\max} + \frac{\Delta V}{2} \right) \Rightarrow K = \frac{2}{\Delta V};$$

$$x_1 = \frac{2}{\Delta V} \left(\lg V - \lg V_{\max} + \frac{\Delta V}{2} \right) = \frac{2 \lg V - \lg V_{\max}}{\lg V_{\max} - \lg V_{\min}} + 1.$$

Перейдем от переменных V, S, Z к переменным x_1, x_2, x_3 .

$$\lg Ag = y = \lg \beta_0 + \beta_1 \lg V + \beta_2 \lg S + \beta_3 \lg Z.$$

Выразим переменные $\lg V, \lg S, \lg Z$ через x_1, x_2, x_3 :

$$(\lg V_{\max} - \lg V_{\min})x_1 = 2 \lg V - 2 \lg V_{\max} + \lg V_{\max} - \lg V_{\min};$$

$$\lg V = \frac{\lg V_{\max} - \lg V_{\min}}{2} x_1 + \frac{\lg V_{\max} + \lg V_{\min}}{2};$$

$$y = \lg \beta_0 + \beta_1 \frac{\lg V_{\max} - \lg V_{\min}}{2} x_1 + \beta_1 \frac{\lg V_{\max} + \lg V_{\min}}{2} +$$

$$+ \beta_2 \frac{\lg S_{\max} - \lg S_{\min}}{2} x_2 + \beta_2 \frac{\lg S_{\max} + \lg S_{\min}}{2} +$$

$$+ \beta_3 \frac{\lg Z_{\max} - \lg Z_{\min}}{2} x_3 + \beta_3 \frac{\lg Z_{\max} + \lg Z_{\min}}{2};$$

$$y = b_0 x_0 - b_1 x_1 + b_2 x_2 + b_3 x_3; \quad x_0 \equiv 1.$$

$$b_0 = \lg \beta_0 + \beta_1 \frac{\lg V_{\max} - \lg V_{\min}}{2} +$$

$$+b_2 \frac{\lg S_{\max} - \lg S_{\min}}{2} + b_3 \frac{\lg Z_{\max} - \lg Z_{\min}}{2}.$$

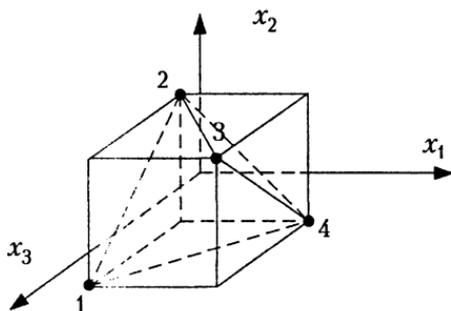
$$b_1 = \beta_1 \frac{\lg V_{\max} - \lg V_{\min}}{2};$$

$$b_2 = \beta_2 \frac{\lg S_{\max} - \lg S_{\min}}{2};$$

$$b_3 = \beta_3 \frac{\lg Z_{\max} - \lg Z_{\min}}{2}.$$

Мы установили однозначное соответствие между переменными V , S , Z и x_1 , x_2 , x_3 , и между параметрами β_0 , β_1 , β_2 , β_3 и b_0 , b_1 , b_2 , b_3 .

В пространстве переменных x_1 , x_2 , x_3 параллелепипед переходит в куб с центром в начале координат и ребром, равным двум.



Фигура, образованная точками 1, 2, 3, 4 называется симплексом. В данном случае имеем правильный симплекс.

Если эксперимент ставить в вершинах куба, то мы имеем так называемый полный факторный эксперимент (ПФЭ) типа 2^n , где n – число факторов.

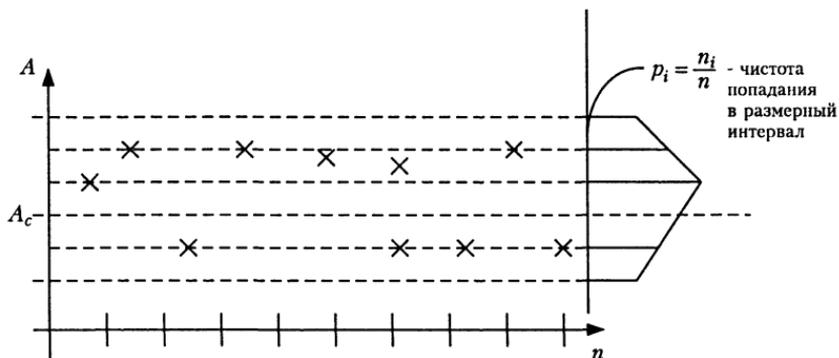
ПФЭ реализует все возможные комбинации уровней независимых переменных, каждая из которых варьируется на двух уровнях. Матрица координат вершин куба называется матрицей плана эксперимента.

При линейном росте числа факторов число опытов растет по показательной функции. Если при решении задачи моделирования можно ограничиться линейным приближением, то ПФЭ неэффективен. Применяется дробный факторный эксперимент. Половина факторного эксперимента называется полурепликой и т.д.

Для нахождения оценок B последовательно находим матрицы:

$$X', X'X, (X'X)^{-1}, (X'X)^{-1}X', (X'X)^{X'Y} = B.$$

Статистический анализ уравнений регрессии. Результаты наблюдений или измерений величины $Y = \{y_1, y_2, \dots, y_N\}$ являются случайными величинами. Действительно, если мы настроим станок на один и тот же размер статической настройки A_C и на одном и том же режиме будем обрабатывать детали, то измеренные размеры деталей будут разными. На точечной диаграмме показаны зависимость размера от ее номера.



Поле рассеяния называется разность $\omega = A_{\max} - A_{\min}$. Разделив ω на K – число интервалов, можно подсчитать количество деталей, попавших в i -й интервал n_i , и статистическую вероятность, или частоту $p_i = \frac{n_i}{n}$.

В реальных задачах n всегда ограничено и называется объемом выборки. Совокупность n измерений называется выборкой из некоторой генеральной совокупности.

Таким образом, наблюдая или измеряя некоторую случайную величину X , мы можем говорить о последовательности наблюдаемых значений x_1, x_2, \dots, x_n , как о совокупности значений одинаково распределенных независимых случайных величин X_1, X_2, \dots, X_n , представляющих n экземпляров одной и той же случайной величины X .

Поэтому все выборочные характеристики – суть функции случайной величины, т.е. они сами случайны.

Например:

$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n} - \text{статистическое среднее}$$

$$\text{или } \bar{X} = \frac{X_1 + X_2 + \dots + X_n}{n}, \text{ т.е. } \bar{x} = \bar{X} - \text{случайная величина.}$$

$$M[\bar{x}] = m_x, D[\bar{x}] = \frac{1}{n^2} S[X_1] + \frac{1}{n^2} D[X_2] + \dots + \frac{1}{n^2} D[X_n] = \frac{D(X)}{n}.$$

$$\text{Статистическая дисперсия } S^2 = \frac{1}{n} \sum (x_i - \bar{x})^2.$$

Если случайная величина X зависит от других случайных или неслучайных величин, например $Y = f(x_1, x_2, x_3)$ или $Ag = f(V, S, Z)$, то зависимость $M[Y] = f(x_1, x_2, x_3)$ называется регрессией случайной величины Y по величинам X_1, X_2, X_3 .

Поэтому мы говорим о регрессионных уравнениях или регрессионных связях. Регрессионные уравнения подвергают статистическому анализу.

Классический статистический анализ состоит из следующих шагов.

1. Делается предположение (выдвигается гипотеза, что наблюдаемая величина Y распределена по нормальному закону).

2. Проверяется гипотеза о нормальности при помощи так называемых критериев согласия. Наиболее часто употребляемые – критерий Пирсона или Колмогорова.

Суть критериев в следующем: сравнивается теоретическое и статическое распределение:

$$\sum_{i=1}^n C_i \left(f(x) - \frac{n_i}{n} \right)^2 = K_{\Pi} - \text{критерий Пирсона.}$$

$p(K_{\Pi} < K'_{\Pi}) = \alpha$, где α – доверительная вероятность, K'_{Π} – пороговое значения критерия, по α находят K'_{Π} и сравнивают с K_{Π} .

Если неравенство выполняется, то говорят, что с вероятностью α (например, 95%) величина X распределена по нормальному закону.

$$\max \left| f(x) - \frac{n_i}{n} \right| = K_K - \text{критерий Колмогорова.}$$

3. В каждой точке факторного пространства делают m параллельных опыта, т.е. реализуется m матриц плана эксперимента.

4. Результаты опытов в каждой точке усредняются.

$$\bar{Y}_g = \frac{1}{m} \sum_{i=1}^m Y_{lg},$$

где Y_{lg} – l -й эксперимент в g -й точке факторного пространства.

$$5. \text{ Вычисляют дисперсию } S_g = \frac{1}{m} \sum_{i=1}^m (Y_{lg} - \bar{Y}_g)^2.$$

6. Проверяют однородность (одинаковость) дисперсий в точках факторного пространства при помощи критерия Кохрена.

$$P\left(G = \frac{S_g \max}{\sum S_g} \leq G_{кр}\right) = \alpha.$$

По α находят $G_{кр}$ и проверяют выполнение неравенства.

7. Рассчитывают коэффициенты B по формуле:

$$B = (x'x)^{-1} x'y.$$

8. Проверяют значимость каждого коэффициента, т.е. выдвигают гипотезу: $|b_i| > 0$. Это делается при помощи t – критерия Стьюдента.

$$P\left(t_i = \frac{|b_i|}{S|b_i|} \geq t_{кр}\right) = \alpha.$$

Распределение Стьюдента t . $\alpha \Rightarrow t_{кр}$.

9. Проверка адекватности модели исследуемому явлению при помощи критерия Фишера.

$$P\left(F = \frac{S_{ад}}{\sum S_g} \leq F_{кр}\right) = \alpha;$$

$$S_{ад} = \frac{1}{N} \sum_{g=1}^N (\hat{Y}_g - \bar{Y}_g)^2.$$

2.2.8. Непрерывные оптимизационные задачи

Назначение оптимального режима обработки, для технологической системы (рис.2.32).

ТС – токарная обработка. Целевая функция – себестоимость.

$$\theta = \frac{C_0}{VS}, \text{ где } C_\theta = \frac{\pi de}{1000} \left(E_C + \frac{E_U}{T} \right);$$

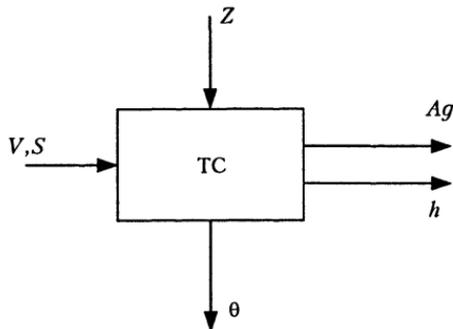


Рис.2.32. Технологическая система как объект управления

$$\left. \begin{aligned} Ag &= \beta_{10} V^{\beta_{12}} S^{\beta_{12}} Z^{\beta_{13}} \\ q &= \beta_{20} V^{\beta_{21}} S^{\beta_{23}} Z^{\beta_{23}} \end{aligned} \right\} \text{уравнения объекта управления.}$$

В партии деталей Z меняется от $Z_{\min} \leq Z \leq Z_{\max}$.
Отсюда

$$Ag_{\max} - Ag_{\min} = \beta_{10} V^{\beta_{12}} S^{\beta_{12}} (Z_{\max}^{\beta_{13}} - Z_{\min}^{\beta_{13}}) = \omega g = Tg,$$

если $\beta_{13} > 0$

$$h = \int_0^t q dt; \quad h_T = \int_0^T q dt \leq h_{\text{доп}} \Rightarrow \beta_{20} V^{\beta_{21}} S^{\beta_{23}} Z_{\text{оп}}^{\beta_{23}} T \leq h_{\text{доп}},$$

где q – скорость износа инструмента.

Запись оптимальной задачи.

$$\theta = \frac{C_{\theta}}{VS} \rightarrow \min;_{V, S}$$

$$V_{\min} \leq V \leq V_{\max};$$

$$S_{\min} \leq S \leq S_{\max};$$

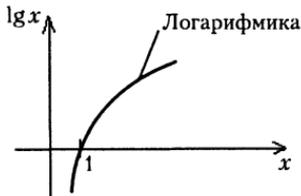
$$\beta_{10} V^{\beta_{11}} S^{\beta_{12}} (Z_{\max}^{\beta_{13}} - Z_{\min}^{\beta_{13}}) \leq Tg;$$

$$\beta_{20} V^{\beta_{21}} S^{\beta_{23}} Z_{\text{оп}}^{\beta_{23}} T \leq h_{\text{доп}}.$$

Логарифмированием можно свести задачу к линейной, относительно логарифмов переменных. (Логарифмирование не сдвигает экстремальные точки функции, например,

$$Y = f(x); \quad \varphi(Y) = \varphi(f(x));$$

если φ – монотонная функция, то и $\frac{d\varphi}{dx} = \frac{d\varphi}{f} \frac{df}{dx}$ и $\frac{d\varphi}{df} > 0$.)



$$\lg \theta = \lg C_{\theta} - \lg V - \lg S;$$

$$\lg V_{\min} \leq \lg V \leq \lg V_{\max};$$

$$\lg S_{\min} \leq \lg S \leq \lg S_{\max};$$

$$\beta_{11} \lg V + \beta_{12} \lg S \leq \lg \frac{Tg}{\beta_{10}(z_{\max}^{\beta_{12}} - z_{\min}^{\beta_{13}})};$$

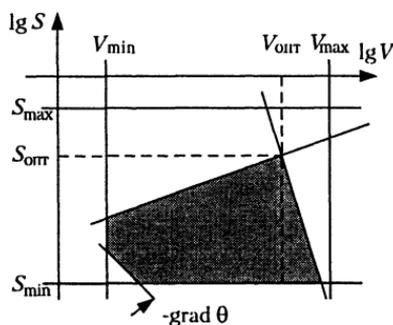
$$\beta_{21} \lg V + \beta_{22} \lg S \leq \lg \frac{h_{\text{доп}}}{\beta_{20} Z_{\text{сп}}^{\beta_{23}}}.$$

Уравнение граничной линии $-\beta_{11} \lg V + \beta_{12} \lg S = a$.

$$\text{Отсюда } \lg S = -\frac{\beta_{11}}{\beta_{12}} \lg V + \frac{a}{\beta_{12}}.$$

Из физического смысла задачи знаки коэффициентов $\beta_{11} < 0$, $\beta_{12} > 0$, $\beta_{21} > 0$, $\beta_{22} > 0$.

Рассмотрим геометрическую интерпретацию задачи линейного программирования. Заметим, что $\text{grad } \lg \theta = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$.



Мы рассмотрели простейший случай однопереходной обработки. В случае многопереходной обработки, количество переменных увеличивается, и между ними возникают связи. Это показывает следующий пример (рис.2.33).

Каждую ТС будем рассматривать как ориентированный объект (см. рис.2.30).

Имеем 6 ТС: $j = \overline{1,3}$; $i_j = \overline{1,2}$.

ТС ji совпадает с припуском z_{ji} .

Запишем выходы и входы всех технологических систем:

$$x_{11}^{\text{ВЫХ}} = 3, \quad x_{12}^{\text{ВЫХ}} = 6, \quad x_{21}^{\text{ВЫХ}} = 5, \quad x_{22}^{\text{ВЫХ}} = 8, \quad x_{31}^{\text{ВЫХ}} = 4, \quad x_{32}^{\text{ВЫХ}} = 7,$$

$$x_{11}^{\text{ВХ}} = 2, \quad x_{12}^{\text{ВХ}} = 4, \quad x_{21}^{\text{ВХ}} = 1 + 3 - 2, \quad x_{22}^{\text{ВХ}} = 5 + 6 - 4, \quad x_{31}^{\text{ВХ}} = 3, \quad x_{32}^{\text{ВХ}} = 6,$$

$$I = 7, \quad II = 8.$$

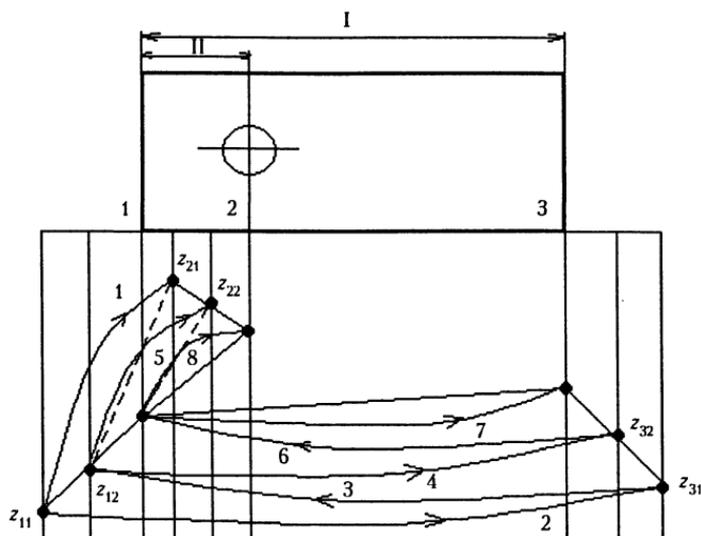


Рис.2.33. Пример связей при многопереходной обработке

Изобразим совокупность ТС виде многомерной системы регулирования (рис.2.34). В нее как бы входят размеры заготовки, выходят размеры детали.

Уравнения системы:

$$x_{11}^{\text{ВЫХ}} = f_{11}(x_{11}^{\text{ВХ}}, \bar{y}_{11}) = f_{11}(2, \bar{y}_{11}) = x_{31}^{\text{ВХ}} = 3;$$

$$x_{31}^{\text{ВЫХ}} = f_{31}(x_{31}^{\text{ВХ}}, \bar{y}_{31}) = f_{31}(f_{11}(x_{11}^{\text{ВХ}}, \bar{y}_{11}) \bar{y}_{31}) = x_{12}^{\text{ВХ}} = 4;$$

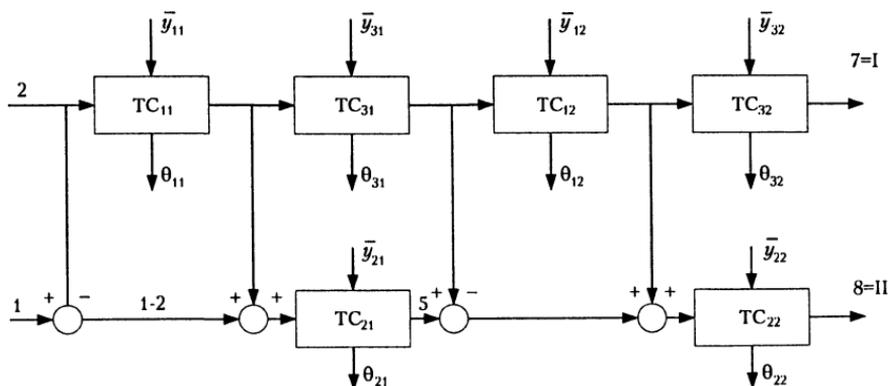


Рис.2.34. Связанная совокупность ТС

$$x_{12}^{\text{ВЫХ}} = f_{12}(x_{12}^{\text{ВХ}}, \bar{y}_{12}) = f_{12}(f_{31}(f_{11}(x_{11}^{\text{ВХ}}, \bar{y}_{11})\bar{y}_{31})\bar{y}_{12}) = x_{32}^{\text{ВХ}} = 6;$$

$$\begin{aligned} x_{32}^{\text{ВЫХ}} &= f_{32}(x_{32}^{\text{ВХ}}, \bar{y}_{32}) = \\ &= f_{32}(f_{12}(f_{31}(f_{11}(x_{11}^{\text{ВХ}}, \bar{y}_{11})\bar{y}_{31})\bar{y}_{12})\bar{y}_{32}) = 7 = I; \end{aligned}$$

$$x_{21}^{\text{ВЫХ}} = f_{21}(x_{21}^{\text{ВХ}}, \bar{y}_{21}) = f_{21}((1 - 2 + f_{11}(2, \bar{y}_{11}))\bar{y}_{21}) = 5;$$

$$\begin{aligned} x_{22}^{\text{ВЫХ}} &= f_{32}(x_{22}^{\text{ВХ}}, \bar{y}_{22}) = \\ &= f_{22} \left(\begin{array}{l} (f_{21}((1 - 2 + f_{11}(2, \bar{y}_{11}))\bar{y}_{21})) - \\ -(f_{31}(f_{11}(x_{11}^{\text{ВХ}}, \bar{y}_{11})\bar{y}_{31})) + \\ +(f_{12}(f_{31}(f_{11}(x_{11}^{\text{ВХ}}, \bar{y}_{11})\bar{y}_{31})\bar{y}_{12})), \bar{y}_{22} \end{array} \right) = 8 = II. \end{aligned}$$

Запись оптимальной задачи.

$\omega_i \leq T_i, \omega_{11} \leq T_{11}$ – ограничения

$\sum_i \sum_j \theta_{ij} \rightarrow \min_{\bar{y}_{ij}}$ – целевая функция.

Решение оптимизационных задач рассматриваемого вида, т.е. оптимизация при наличии ограничений – неравенств, называется задачей математического программирования.

Решению задач математического программирования посвящена многочисленная литература.

Литература

- Автоматизированные системы технологической подготовки производства в машиностроении / Под ред. Г.К.Горанского. М.: Машиностроение, 1976.
- Вуши Г.* Теория систем, М., 1978
- Гетманова А.Д.* Логика. М.: Добросвет, Университет, 1998.
- ГОСТ 234.003-90 / Автоматизированные системы. Термины и определения.
- Зыков А.А.* Основы теории графов. М.: Наука, 1987.
- Информационные технологии в промышленности и экономике: Сб. науч. трудов ИКТИ РАН / Под ред. Ю.М.Соломенцева. М.: Янус-К, 2001. Вып.№3.
- Кофман А.* Введение в прикладную комбинаторику. М.: Наука, 1975.
- Кофман А.* Введение в теорию нечетких множеств. М.: Радио и связь, 1982.
- Кофман А., Апри-Лабордер А.* Методы и модели исследования операций. М.: Мир, 1977.
- Куратовский К., Мостовский А.* Теория множеств. М.: Мир, 1970.
- Марка Д., Мак Гоуэн К.* Методология структурного анализа и проектирования / Пер. с англ. М., 1993.
- Гусев А.А., Павлов В.В., Андреев А.Г.* и др. Технология сборки в машиностроении. Т. III-5 // Машиностроение: Энциклопедия / Под общ. ред. Ю.М.Соломенцева. М.: Машиностроение, 2001.
- Павлов В.В.* Математическое обеспечение САПР в производстве летательных аппаратов. М.: МФТИ, 1978.
- Павлов В.В.* Полихроматические графы в теории систем // Информационные технологии. 1998. №6. С.2-9.
- Павлов В.В.* Полихроматические множества в теории систем. Операции над PS-множествами // Там же. 1998. №3. С.8-13.
- Павлов В.В.* CALS-технологии в машиностроении: (математические модели) / Под ред. Ю.М.Соломенцева. М.: ИЦ МГТУ СТАНКИН, 2002.
- САПР. Типовые математические модели объектов проектирования в машиностроении // Методические указания РД 50-464-84. М.: Стандарты, 1985.
- Соломенцев Ю.М., Павлов В.В.* Моделирование технологической среды машиностроения. М.: Станкин, 1994.
- Таха Х.* Введение в исследование операций. М.: Мир. 1975. Т.1-2.
- Фундаментальные физико-математические проблемы и моделирование технико-технологических систем: Сб. научных трудов / Под ред. Л.А.Уварова. М.: Станкин, 2000.

Особенностью научно-технического прогресса последнего времени является повсеместное использование персональных электронно-вычислительных машин (ПЭВМ) в самых различных сферах производственной и управленческой деятельности. Это связано с тем, что только в последнее время компьютерная техника стала доступна для применения конкретными работниками на отдельных рабочих местах. С точки зрения пользователя применение ПЭВМ резко изменяет привычный облик многих профессий за счет:

- устранения утомляюще-однообразного характера выполняемой работы;
- возможности предоставления большего времени для творческого труда;
- интенсификации интеллектуальной деятельности;
- уменьшения числа ошибок в ходе практической деятельности;
- повышения качества конечных результатов;
- накопления и применения повторно-используемых компьютерных баз знаний.

Все разнообразие применения ПЭВМ в производственных условиях опирается на возможности определенным образом спроектированного базового и прикладного программного обеспечения. Совокупность такого программного обеспечения образует новую разновидность технологической среды для выполнения той или иной профессиональной деятельности. Массовое распространение персональных компьютеров привлекло на рынок программных продуктов огромное количество пакетов прикладных программ (ППП)

различного назначения. В начале 1990-х годов большие надежды возлагались на текстовые и графические редакторы, на программы, автоматизирующие отдельно взятые проектные процедуры и операции в ходе жизненного цикла изделия. На основе этого факта возникло мнение, что обилие существующих ППП позволит решить проблему автоматизации деятельности пользователей предприятия с помощью поставляемых в коробках типовых решений от фабрик по программированию (так называемые коробочные решения). На практике это мнение получило подтверждение лишь отчасти. Это связано с двумя причинами.

Во-первых, упомянутые средства не позволяли автоматизировать весь процесс в целом, они лишь механизировали его, и в то же время каждая из отдельных программ требовала собственного ввода исходных данных (зачастую в больших объемах). Причем эти данные не были согласованы между различными проектными процедурами. Сквозной автоматизации не получалось. Повышение производительности труда проектировщиков и частичное сокращение сроков выдачи проектной документации достигались за счет распараллеливания процесса проектирования и привлечения большого числа проектировщиков. А распараллеливание осуществлялось исходя из возможности декомпозировать проектируемую систему на относительно автономные, функционально завершённые изделия, узлы и агрегаты, подсистемы снабжения, задачи контроля и управления и т.д. Сроки проектирования действительно сокращались, но между группами проектировщиков как одной, так и разных специальностей с неизбежностью стали возникать дополнительные потоки информации и промежуточные документы, необходимые лишь для согласования и уточнения. Это осложнилось общей итерационностью процесса проектирования.

Во-вторых, информационная потребность конечного пользователя столь неоднозначна и динамична, что простое применение одного или нескольких интегрированных ППП не дает гарантии рационального использования персональной вычислительной техники на конкретном рабочем месте в рамках организации деятельности всего предприятия.

Научная проблематика автоматизации конструкторско-технологической подготовки производства (КТПП) в машиностроении традиционно носит двойственный характер:

- разработка удобных способов спецификации технических процессов, охватывающих основные виды деятельности; обычно этот круг задач решается в рамках информационного анализа прикладной области;

- создание уникальных и адаптация универсальных программных систем общего назначения для решения конкретных задач (операционные системы, базы данных, машинная графика, сети ЭВМ и т.д.); эти решения обычно рассматриваются собственно при построении автоматизированных систем.

Большая часть разработок, посвященных созданию автоматизированных систем КТПП, фиксирует основное внимание на алгоритмах, машинной графике, базах данных и т.д. Однако при этом зачастую теряется долгосрочная цель разработки автоматизированной системы – эффективность КТПП в ходе всего жизненного цикла создания и эксплуатации изделия. При постоянной борьбе за рынки сбыта комплексным показателем эффективности КТПП может выступать конкурентоспособность выпускаемой продукции. Практика показывает, что лидирующие позиции на рынках сбыта могут быть достигнуты только при соблюдении следующих условий:

- совершенствования управления и 100% контроля за деятельностью всего коллектива производящего предприятия;
- интеграции в ходе всего жизненного цикла отдельных видов деятельности по созданию и подготовке производства изделия на базе информационных технологий;
- обеспечения средствами информационных технологий возможности для максимального раскрытия и усиления творческого потенциала конечных пользователей, действия которых собственно и определяют успех или неудачу новой разработки в целом.

Все это требует нового взгляда на формализованное описание собственно процесса профессиональной деятельности, подлежащей автоматизации, на архитектуру построения прикладных приложений, на построение интерфейсов взаимодействия пользователей с компьютерной средой и, наконец, инструментальных программных средств, которые способны обеспечивать повседневную деятельность пользователя в компьютерно-технологической среде с минимальным привлечением профессиональных программистов.

Именно это является обоснованием необходимости комплексного методологического подхода к решению задач построения автоматизированных систем КТПП (рис.3.1), которые наиболее полно представлены в CALS-технологии. Назначение этой методологии призвано помочь модернизировать и совершенствовать в первую очередь информационную и организационную структуру машиностроительного предприятия таким образом, чтобы отслеживались и решались базовые проблемы его развития, предопределяющие стабильные успехи не только в настоящем, но и в будущем. Эта методология существенно опирается на современные научно-технические достижения, включая

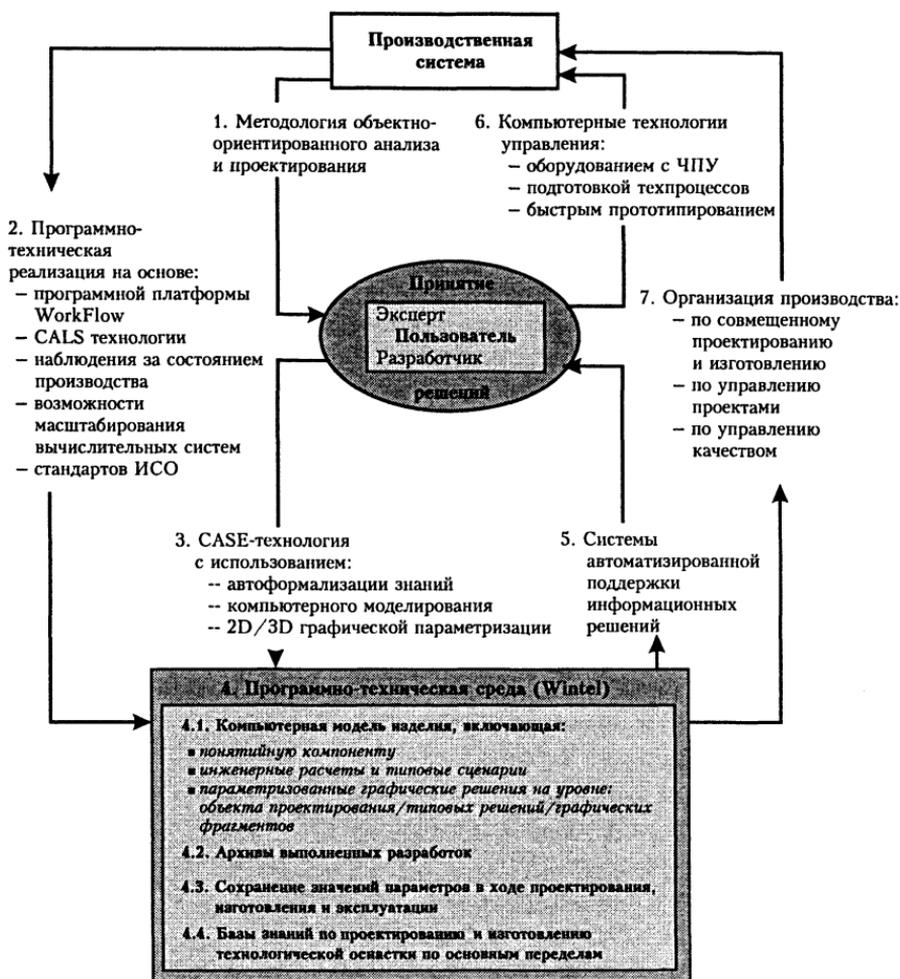


Рис.3.1. Комплексный подход к решению задач производящего предприятия при использовании информационных технологий

объектно-ориентированный подход, CASE и Workflow-технологии, стандарты ИСО, гипертекстовые и экспертные системы и т.д.

Исходя из этих подходов, в данной работе сделана попытка обосновать выбор новой архитектуры построения автоматизированной системы КТПП – систем автоматизированной поддержки информационных решений (САПИР), показать их роль и место в производственной

структуре предприятия на примере отечественных инструментальных программных средств и опыта разработок в промышленности.

Общую цель автоматизации деловых процессов в рамках компьютерно-технологической среды можно определить как необходимость более эффективного совершенствования управлением деятельностью производящего предприятия (КБ + завод) в условиях рынка. Чтобы компьютерно-технологическая среда отвечала этой глобальной цели, непосредственные участники создания последней должны усвоить и точно понять потребности протекающих на предприятии деловых процессов и сотрудников их исполняющих, т.е. разработать модель деятельности производящего предприятия. Затем на этой основе проектируется и реализуется компьютерно-технологическая среда, которая будет адекватно отвечать конечным нуждам производящего предприятия и сможет адаптироваться собственными силами в течении 5–10 лет. Этапность по переводу традиционной деятельности к автоматизированному варианту представлена на рис.3.2.

Концептуальная модель перехода от традиционного проектирования к компьютерной среде приведена на рис.3.3. Данная модель предусматривает предварительный анализ деятельности специалистов при решении задач традиционным способом в форме деловых (технических) процессов (левая часть рис.3.3). Как показывает практика в машиностроении все многообразие действий специалистов можно определить через следующие базовые типы деловых процессов:

- заполнение и редактирование данных в диалоге;
- выбор данных из таблиц;
- выполнение расчета;
- создание и редактирование графического решения;
- выбор варианта действий из списка допустимых;
- сохранение промежуточных результатов;
- просмотр и утверждение результатов.

Деловые процессы по проектированию в машиностроении достаточно успешно автоматизируются в рамках программной платформы под названием интегрированная интеллектуальная система ИИС (правая часть рис.3.3). В МГТУ «Станкин» достаточно давно (первые работы относятся к 1984 г.) используется объективно-ориентированная технология ИИС для проектирования на основе моделей. Символьные и графические базы знаний соответственно создаются с помощью инструментальных средств интеллектуальная компьютерная среда (ИКС) и T-FLEX CAD, и поддерживают создание «строительных кирпичиков» проектирования и «разумных деталей», включая электродвигатели, штампы, пресс-формы, режущие и измерительные инструменты и т.д. Таким образом, ИКС, ИИС и T-FLEX

Анализ традиционной деятельности и предметной области

Компоненты, обеспечивающие автоматизацию деятельности на ПЭВМ

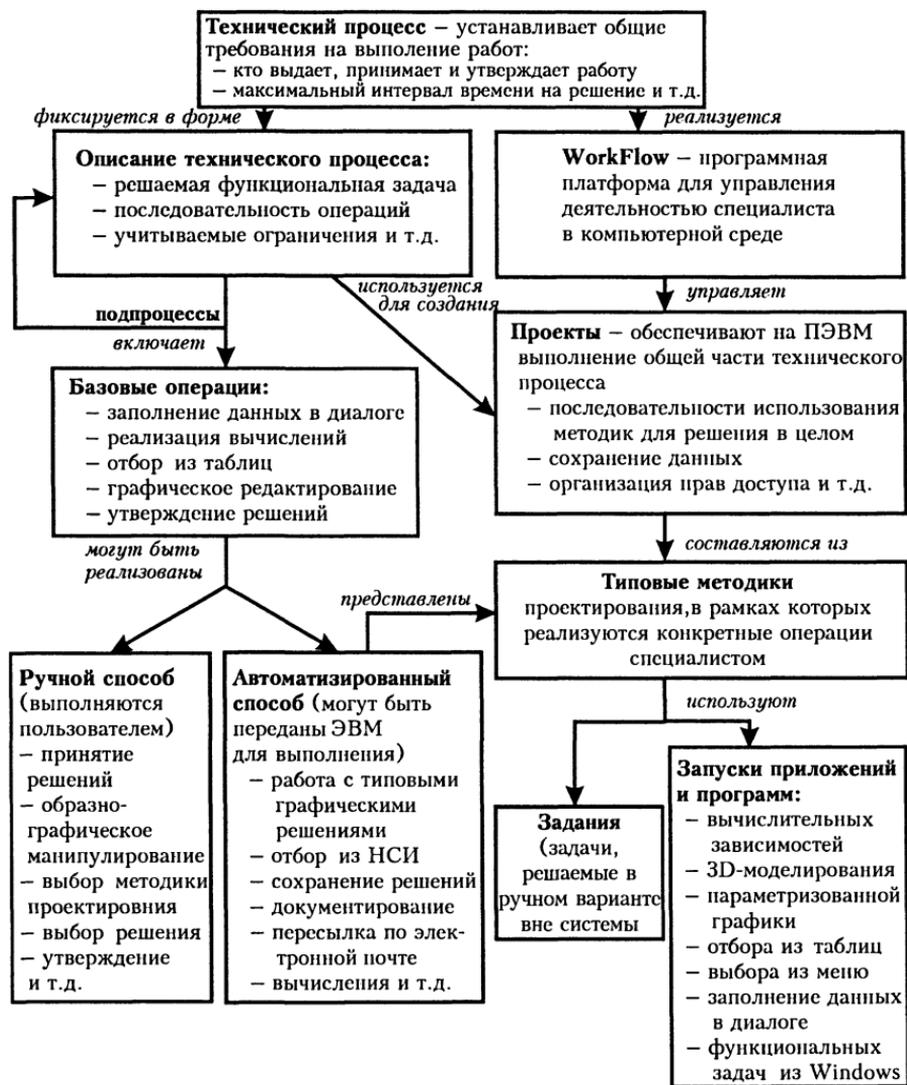


Рис.3.3. Концептуальная модель перевода традиционной деятельности при проектировании в машиностроении к автоматизированному варианту

- применение концептуальной модели перевода традиционной деятельности специалиста к компьютерному варианту с минимальным участием программистов;
- использование формы представления знаний для описания правил проектирования как в символьном, так и в графическом вариантах, максимально приближенной к сложившейся в предметной области;
- использование объектно-ориентированного проектирования на основе моделей, при этом не исключая возможности по созданию чертежно-конструкторской документации с применением традиционных средств ПЭВМ.

2. Новые архитектурные принципы построения САПИР. Эти архитектурные принципы призваны обеспечивать:

- использование стиля программирования, ориентированного на управление событиями в мире пользователя, а не разработчика программных систем;
- использование единой программной платформы типа Workflow, структуры и способа взаимодействия между отдельными программными компонентами при учете основных достоинств семейства Windows 95/NT;
- единое представление делового процесса;
- единое построение компьютерной базы знаний для решения задач машиностроения;
- единая структура представления «Проекта» на создание нового объекта (Архив, ФТЗ – формализованное техническое задание, Расчет, Конструкция, Технология);
- реализацию совмещенного во времени проектирования основного изделия и компьютерной подготовки его производства;
- возможность силами пользователей создавать, адаптировать и наращивать прикладные информационные системы за счет CASE-средств.

К особенностям разработки САПИР, помогающих пользователю в профессиональной деятельности, можно также отнести следующее:

- возможность адаптации автоматизированных систем к личности исполнителя и интересам предприятия («настройка на пользователя»);
- учет особенностей и ограничений человеческой системы переработки информации.

Использование САПИР может помочь повысить качество и резко снизить время, требующееся для промышленной разработки нового объекта в компьютерной среде. При этом предприятие все в большей и большей степени становится клиенто-ориентированным, т.е.

отвечающим нуждам потребителей. Использование САПИР также позволяет изменить текущий фокус инженерной деятельности конструктора при работе с ПЭВМ от ныне существующих 70% рутинной работы и 30% созидательной работы на прямо противоположное: 70% созидательной и 30% рутинной.

Идеи, методики и инструментальные средства, представленные в настоящей книге, были проверены на практике в ходе работ по заказу Министерства промышленности, науки и технологий РФ на ОАО «Сафоновский электромашиностроительный завод». Внедренные в производственные условия компоненты приведены на рис.3.4.

| Слой | Стадии | | | |
|------------------------|---|--|---|--|
| | Проектирование изделия Управление проектами | Производство изделия | | Эксплуатация изделия |
| ТПП | | Изготовление изделия | | |
| 1. Организационный | Проектирование изделия Управление проектами | Производство изделия | | Эксплуатация изделия |
| 2. Информационный | Научные исследования и конструирование. Формирование ЕИПП и БЗ | Технологическое проектирование. Формирование БЗ на оснастку | Оценка технологичности и эффективности производства | Оценка эффективности эксплуатации изделия. Сохранение КД. |
| 3. Управления качества | Контроль качества при проектировании | Контроль качества при технологическом проектировании | Контроль качества при изготовлении | Контроль качества в ходе эксплуатации |
| 4. Модельный | Компьютерное моделирование изделия: – функциональных свойств; – механических свойств и т.д. | Компьютерное моделирование процессов ТПП: – штамповки; – литья пластмасс; – механо-обработки и т.д. | Компьютерное моделирование производственной системы на уровне: – участка; – цеха; – завода | Компьютерное моделирование эксплуатации изделия Рекомендации по повышению потребительских свойств изделия |
| 5. Материальный | Оформление комплекта конструкторской документации (КД) на изделие | Изготовление технологической оснастки | Изготовление изделия | Эксплуатация, техническое обслуживание и ремонт |

Рис.3.4. Компоненты проектных и производственных работ, выполняемых в ходе реализации изделия в компьютерной среде

Авторы благодарны студентам МГТУ «Станкин» за участие в обсуждении настоящих материалов в ходе лекционных, практических и лабораторных занятий.

3.1. Место и роль информатики в индустриальном обществе (на примере задач конструкторско-технологической подготовки производства)

На наших глазах наука добилась поразительных успехов и стала мощной силой, формирующей жизни миллионов людей на планете. В 60 годы XX столетия роль и место эволюционного развития науки была показана Т.Куном с использованием понятия парадигмы. Под парадигмой у Т.Куна в широком смысле понимается набор убеждений, теорий, научных ценностей и техник, разделяемых большинством представителей научного сообщества. В настоящей работе сделана попытка выявить те изменения в множестве общенаучных феноменов, которые вызваны информатизацией общества. И на основе этих изменений попытаться показать место и роль информатики в индустриальном обществе на ближайшие годы. Демонстрация возможностей информационных технологий проводится на примере частной задачи – конструкторско-технологической подготовки производства в машиностроении (КТПП).

Исторически сложилось так, что бизнес в области информатики сильно монополизирован. Поэтому за той или иной парадигмой развития ВТ обычно стоит вполне конкретная фирма-производитель. И в этой ситуации необходимо достаточно полно, своевременно и объективно представить информацию для руководителя о новых тенденциях, связанных с информатизацией технологической среды при решении задач КТПП.

Внешними признаками парадигмы при КТПП на нынешний момент выступают:

- наличие общности ценностей (сохранение биосферы, применение ресурсосберегающих технологий, учет экологических последствий производства, проверяемость, воспроизводимость, логическая непротиворечивость, наглядность или приемлемый допуск ошибки и т.д.);
- использование символических обозначений (применение сложившейся математической нотации, практика выражения определенных устойчивых связей в кратких уравнениях типа $D = R^* 2$, фиксация стандартов на оформление результатов отдельных видов деятельности (ЕСКД для формирования чертежей и ЕСПП для представления техпроцессов и т.д.));
- наличие специфического языка «деловой прозы» при записи исходных данных и ограничений, фиксации решения, оформлении результатов деятельности в той или иной прикладной области; специалист должен понимать и выражаться на этом языке;

- доверие к устоявшимся частным моделям (книга – источник знаний, конвейер – способ организации деятельности группы людей при выпуске массовой продукции, чертеж – язык техники и т.д.);
- моделирование, т.е. когда решение конкретной задачи подменяется другим; (например, вместо натурального испытания технической системы реализуется компьютерное моделирование); но это другое решение предоставляет исследователю возможность анализировать свойства, существенные при решении конкретной задачи (использование метода конечных элементов для компьютерного моделирования технологических процессов в штамповке, при проектировании и изготовлении пресс-форм и т.д.).

Насыщение производственных процессов в машиностроении средствами вычислительной техники наиболее существенно отразилось на трех последних признаках парадигмы.

Согласно Т.Куну, парадигмы играют в истории науки решающую, сложную и неоднозначную роль. На определенных стадиях развития они действуют как концептуальная смирительная рубашка, т.е. «покушаются» на возможности новых открытий и исследования новых областей реальности. В истории науки прогрессивная и реакционная функции парадигм чередуются с некоторым предсказуемым ритмом.

Новая, радикальная теория никогда не будет дополнением или приращением к существующим знаниям. Она меняет основные правила, требует решительного пересмотра или переформулирования фундаментальных допущений прежней теории, проводит переоценку существующих фактов и наблюдений. Сдвиги от аристотелевской к ньютоновской физике или от ньютоновской к эйнштейновской, от геоцентрической системы Птолемея к астрономии Коперника и Галилея, или от теории флогистона к химии Лавуазье – замечательные примеры изменений этого рода. В информатике аналогичным примером служит переход от спорадического программирования к структурному и далее к объектно-ориентированному, резкое увеличение доли распределенных систем над централизованными, переход от отдельных автоматизированных рабочих мест к интеграции технологических процессов на компьютерной основе, вплоть до создания расширенных (виртуальных) предприятий с элементами CALS-технологий и т.д.

Во время переходного периода от старой парадигмы к новой встает проблема коммуникации. Представители разных школ оперируют разными базовыми постулатами о природе реальности и по-разному определяют элементарные понятия. Научные критерии

разнятся, аргументы зависят от парадигмы, а осмысленная кофронтация невозможна без взаимной интерпретации понятий. Коммуникация через концептуальную перегородку будет заведомо неполной и приведет к путанице. В качестве характерного примера можно привести полное различие по смыслу таких понятий, как материя, пространство и время в ньютоновской и эйнштейновской моделях.

В табл.6 приведены особенности информационной среды, используемой в период перехода от старой технологии к новой.

Таблица 6

Особенности использования и перехода к новой среде представления

- В начале перехода к новой среде представления ее возможности воспринимаются пользователем в терминах старой среды, новые возможности проявляются при решении традиционных задач (напомним, что автомобиль при зарождении называли «Безлошадный экипаж»)
- Новая среда представления решение некоторых задач облегчает, а некоторых – усложняет
- В рамках среды представления возникают три типа понимания и общения со средой: я – ты – он. Опыт человека
- Любая среда представления отражает достигнутый уровень осознанности пользователем прикладной задачи с некоторой, вполне определенной нечеткостью
- Среда представления влияет на образ мыслей пользователей. («Часы» обеспечивают не только измерение продолжительности отрезков времени, но и «синхронизируют» деятельность общества). Влияние среды представления выражается через следующие характеристики:
 - Совокупность базовых категорий, принятых в среде, в сильной степени определяет всю структуру описания прикладной области, методы исследования и работы с ней
 - Большие структуры обладают большой инерцией («Камень на лужайке»)
 - Влияние понятийного уровня. В том числе и использование национального языка и традиций
 - При общении с компьютерной базой знаний нельзя позволить себе роскошь «туманности»
 - Предшествующий практический опыт по взаимодействию со средой представления может дать как положительный, так и отрицательный эффект

В последнее время на наших глазах происходит переход от пассивного восприятия к активному воздействию человека на мир. Именно этот переход привел к резкому росту роли и значимости информации в современном мире и в особенности в той его сфере, где производятся материальные ценности. Здесь следует отметить, что массовым носителем знаний при пассивном восприятии мира выступают книги, журналы, газеты и т.д., а во втором – средства информационных технологий и телекоммуникационной техники. В ведущих

странах мира разработаны долговременные проекты и активно ведутся работы по созданию компьютерных баз знаний и корпоративных информационных систем для реализации товаров и услуг на принципах «Бизнеса в высоком темпе» (Commerce At Light Speed – является одной из возможных расшифровок аббревиатуры CALS).

Суть «Бизнеса в высоком темпе» связана с переходом от технологий, основанных на интенсивном индивидуальном труде по созданию изделий, удовлетворяющих специфическим требованиям одного конкретного применения, к технологиям, основанным на планируемых капиталовложениях в разработку повторно – используемых агрегатов и компонентов. Эти технологии во многом базируются на достижениях средств ВТ и телекоммуникации по созданию распределенных баз данных и знаний и внедрению компьютерного моделирования изделия. Особенно эффективны эти технологии при организации производства серий функционально-подобных продуктов, составляемых из компонентов и агрегатов и максимально учитывающих текущие запросы пользователя.

Особенности наблюдаемого технологического перехода кратко можно сформулировать следующим образом:

- *целью деятельности промышленного предприятия является постоянное совершенствование качества продуктов и услуг (в отличие от традиционного повышения производительности «любой ценой»); критерии качества на продукты и услуги задает конечный потребитель; в центр внимания ставится не числовой показатель результата той или иной производственной функции или деятельности, а качество процесса ее выполнения;*
- *внедрение принципа «один раз создать и многократно использовать» за счет создания и накопления баз данных и знаний в компьютерной форме; эта же основа призвана обеспечить совмещенное проектирование основного изделия и компьютерной подготовки его производства.*
- *повышение роли решений и инициативы каждого исполнителя; деление на производственные подразделения носит условный характер, так как поощряется бригадная работа для достижения конечных результатов;*
- *организация работ предприятия должна динамично трансформироваться и совершенствоваться для достижения поставленных целей деятельности; для этого исследуются и фиксируются деловые процессы (бизнес-процессы), на основе которых устраняются недостатки производственной системы, а не отдельных работников;*

- *гибкость и живучесть производственной системы* во многом обеспечивается накоплением компьютерной базы знаний по техническим процессам и их интеграцией на основе информационной технологии в ходе всего жизненного цикла создания и производства продуктов и услуг.

Как побочный, а не главный результат всего этого технологического перехода является снижение затрат на производство. Именно поэтому CALS-технологии рассматриваются как стратегия промышленности и правительства, направленная на эффективное создание, обмен, управление и использование электронных данных, поддерживающих Жизненный Цикл Изделия с помощью международных стандартов, реорганизации предпринимательской деятельности и передовых технологий.

3.1.1. Новое понимание реальности и существования в условиях информационных технологий

Внешнее влияние парадигмы при КТПП можно проанализировать через взаимосвязанную систему форм представления знаний (ФПЗ) об окружающем мире, способах познания (ФПЗ1), накопления знаний (ФПЗ3, ФПЗ6) и обучения подрастающего поколения (ФПЗ4), способах воздействия на окружающий мир (ФПЗ2, ФПЗ5). На основе этих взаимосвязей можно сделать попытку определения роли и места человека и информатики в этом мире (рис.3.5) и в первую очередь в технологической среде.

Далее более подробно остановимся на изменении форм представления знаний при переходе к индустриальному обществу и насыщению производственной среды средствами вычислительной техники.

Замечено, что сложность решения задач существенно зависит от формы представления знаний. Продемонстрировать этот факт можно на примере следующих записей:

- 1) $2 + 9 = 11$ – арабская система счисления;
- 2) $II+IX=XI$ – римская система счисления.

Как в арабской, так и в римской системе счисления использованы одни и те же исходные данные (два и девять), выполнено одно и то же действие (сложение), получен одинаковый результат (одиннадцать). Но если, например, ребенок в пяти-семилетнем возрасте с успехом может освоить базовые алгоритмы всех действий арифметики в арабской системе счисления (взять два числа, записать их в столбик и т.д.), то эти же алгоритмы в римской системе счисления зафиксировать, а значит и обучить им подрастающее поколение, представляется весьма затруднительным. В одной из публикаций указывается, что данный факт мог стать одной из причин упадка и краха Великой Римской империи. *(Невозможность за приемлемое время*

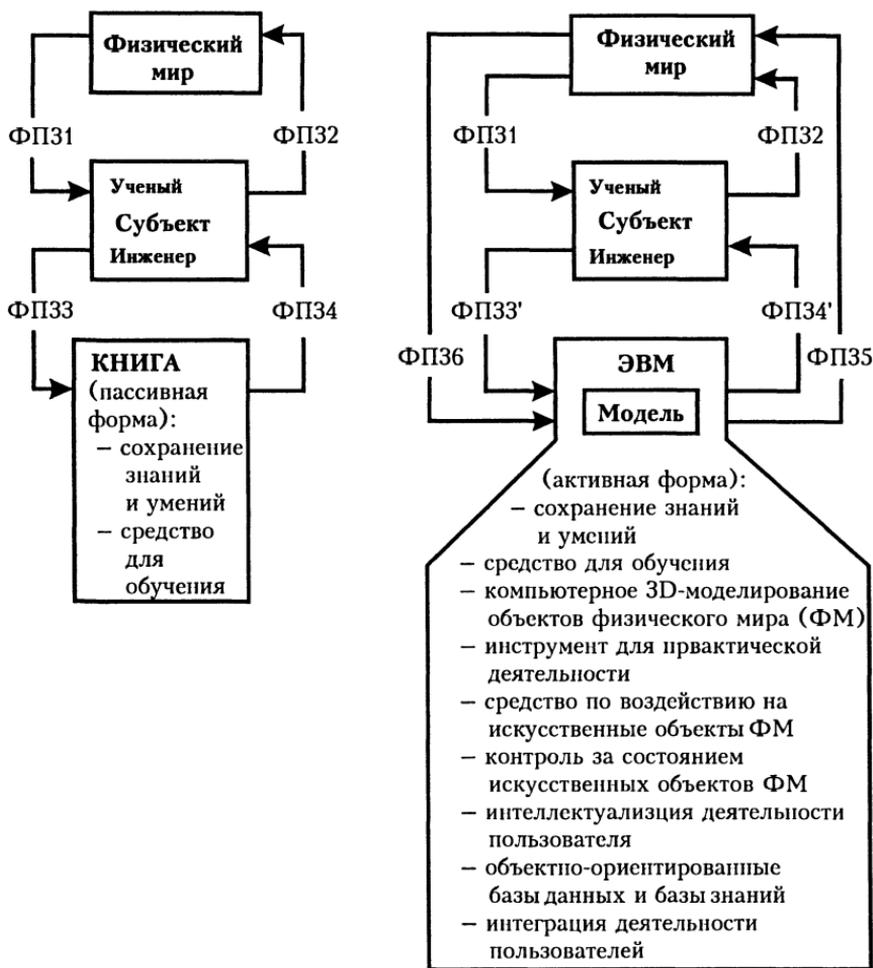


Рис.3.5. Роль и место ЭВМ в индустриальном обществе.
ФПЗ – сокращение от словосочетания «форма представления знаний»

воспроизводить новое кадровое пополнение для поддержания жизнедеятельности технологической среды с учетом растущих потребностей окружающего мира.)

По нашему мнению различие между традиционной технологией (рис.3.5а) и новой информационной технологией (рис.3.5б) напрямую связано с появлением и массовым внедрением ПЭВМ и средств ВТ в рамках технологической среды в машиностроении. Основные моменты, связанные с возможностями применения новой информационной технологии (НИТ) в машиностроении, представлены в табл.7 в следующей форме:

Таблица 7

Роль новой информационной технологии при реорганизации деятельности машиностроительного предприятия

| | |
|--|---|
| 0. Компьютерная технология | Традиционный стиль решения задач |
| 1. Создание средств ВТ для работы в промышленности | Возможный вариант решения в условиях НИТ |
| 2. Компьютерное управление качеством | Ручное управление средствами производства |
| 3. Корпоративная информационная система | Компьютерно-интегрированное производство |
| 4. Компьютерная модель объекта | Качество производимой продукции контролируется периодически и выборочно |
| 5. Компьютерное моделирование | Качество контролируется оперативно, ведется мониторинг за деятельностью основных производственных процессов |
| 6. Технология баз знаний | Последовательная организация работ по созданию объекта и подготовки его производства на основе функциональных подразделений |
| 7. Экспертные системы | Параллельно – последовательная организация работ по созданию объекта и подготовки его производства |
| 8. Средства поддержки принятия решений | Чертеж – язык техники |
| 9. Сети ЭВМ | Твердотельная модель – основа для всего жизненного цикла объекта от момента его создания до уничтожения. |
| 10. Технологии автоматического индексирования и отслеживания | Конечные свойства объекта устанавливаются в ходе натурных испытаний |
| | Большая часть сведений о свойствах объекта может быть получена при компьютерном моделировании до организации выпуска изделия. |
| | Пассивная форма сохранения знаний о ранее выполненных разработках (архив, стандарты предприятия (СТП), библиотеки типовых решений и т.д.) |
| | Накопление «ноу-хау» предприятия по принципу «один раз создать и все могут многократно использовать в своих разработках» |
| | Сложную работу могут выполнять только эксперты предметной области |
| | Большую часть работы могут выполнять специалисты по общим вопросам, используя накопленный опыт экспертов по решению задач |
| | Все решения принимают менеджеры |
| | Принятие решений становится частью работы каждого сотрудника (иерархическое принятие решений) |
| | Специалистам для получения, хранения, поиска и передачи данных требуется офис |
| | Специалисты могут посылать и получать информацию из того места, где они находятся. Повышается оперативность и наглядность данных |
| | Для того чтобы найти некую сущность, необходимо знать, где она находится |
| | Сущности говорят вам, где они находятся |

В общем случае жизненный цикл изделия от конструкторского замысла до промышленного воплощения и набор операций, его обеспечивающий, приведен на рис.3.6. Здесь в квадратах указаны отдельные контролируемые состояния, а линии обозначают операции, с помощью которых происходит переход из одного состояния в другое. В рамках данного жизненного цикла центральное место естественно отводится состоянию «*Описание изделия, зафиксированное в конструкторской документации, Оч*», т.е. представлению нового изделия в виде сборочных и детализированных чертежей, спецификаций, расчетно-пояснительных записок и т.д. На практике данный жизненный цикл реализуется в соответствии с рис.3.7. Обычно в рамках конструкторского бюро оформляется чертежно-конструкторская документация на изделие, а на заводе происходит собственно реализация последнего. Активными компонентами при такой организации выступают люди. На рис.3.7 компоненты, определяющие качество конечного изделия, выделены в виде овала.

При традиционной технологии основные операции, с помощью которых реализуется и фиксируется проектируемое изделие, выполняются с помощью средств по созданию двухмерных чертежей. В нашей стране создание чертежей должно отвечать требованиям единой системы конструкторской документации (ЕСКД). Набор стандартов ЕСКД направлен на упорядочение деятельности человека, так как



Рис.3.6. Жизненный цикл изделия от конструкторского замысла до воплощения и набор операций, его обеспечивающий.

Здесь: **НФО** – набор формализованных операций подготовки чертежа (ЕСКД); **ОВЗ** – операции выбора заготовки; **ОПС** – операции, реализуемые производственной системой; **КСИ** – критерии соответствия изделия требованиям чертежа; **КСД** – критерии соответствия требований чертежа замыслу конструктора



Рис.3.7. Традиционная организация проектирования и изготовления изделия в машиностроении

создание и чтение чертежей выполняется именно им. Основным носителем чертежа выступает бумага. Для простоты и удобства работы человека с ЕСКД принята серия умолчаний, т.е. на чертеже часть информации отсутствует и не отображается. Поэтому простое сканирование чертежей и перевод их в электронную форму вовсе не аналогичен созданию компьютерной модели изделия.

Такая роль чертежно-конструкторской документации вполне понятна, так как до середины 1980-х годов человечество практически не имело в своем распоряжении ни каких других инженерных способов фиксации и передачи сведений о вновь создаваемом изделии кроме чертежей. Только с появлением компьютера и внедрением 3D графики стало ясно, что чертеж является одной из форм представления знаний о вновь проектируемых изделиях.

Новая схема организации производства при использовании информационной модели изделия приведена на рис.3.8. Эта схема призвана обеспечить совмещенное проектирование основного изделия и компьютерную подготовку его производства. Подробнее совмещенная схема организации производства на примере электродвигателя рассмотрена в параграфе 3.7.

Новый взгляд на место и роль информатики в индустриальном обществе нашел свое отражение и в реорганизации практической деятельности. Наиболее существенные изменения, связанные с

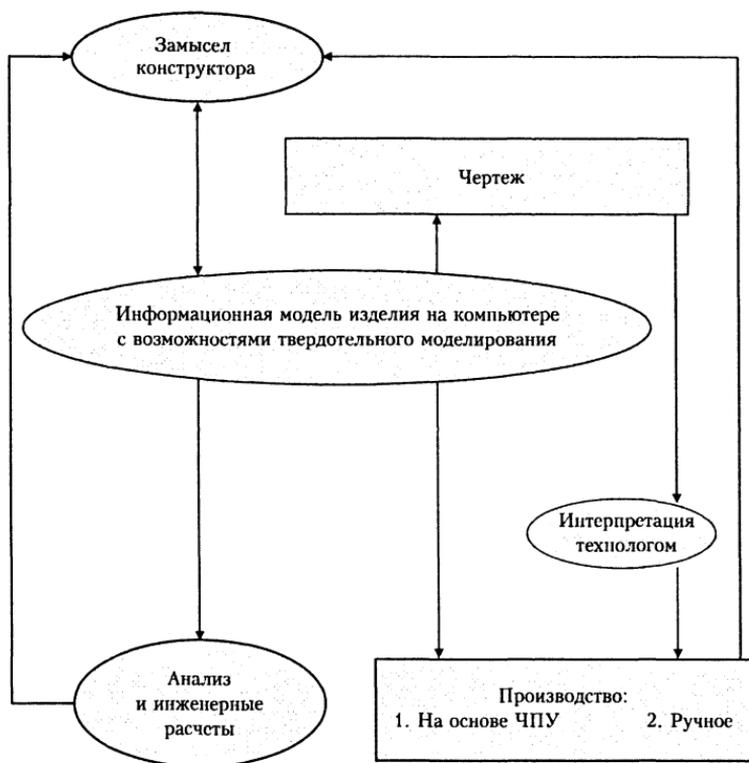


Рис.3.8. Организация совмещенного проектирования и изготовления изделия в машиностроении

переосмысливанием места и роли информатики в индустриальном обществе, коснулись также и учебного процесса. Это обусловлено тем, что сложившаяся методология обучения студентов в машиностроении ориентирована на пассивную передачу «готовых» знаний «впрок». Традиционный процесс обучения строится с претензией на то, чтобы передаваемых знаний хватило едва ли не на все возможные случаи будущей учебной или трудовой деятельности. Стержнем этой методологии выступает учебник. (Вот почему изобретение книгопечатания революционизировало образовательный процесс.)

В современных условиях профессиональная подготовка специалистов строится на активизации познавательной деятельности студентов, чтобы весь учебный процесс был пронизан проблемностью и эвристичностью. Нужно учить студентов думать. (Важен не только результат, но и метод, с помощью которого он достигается.) Эта

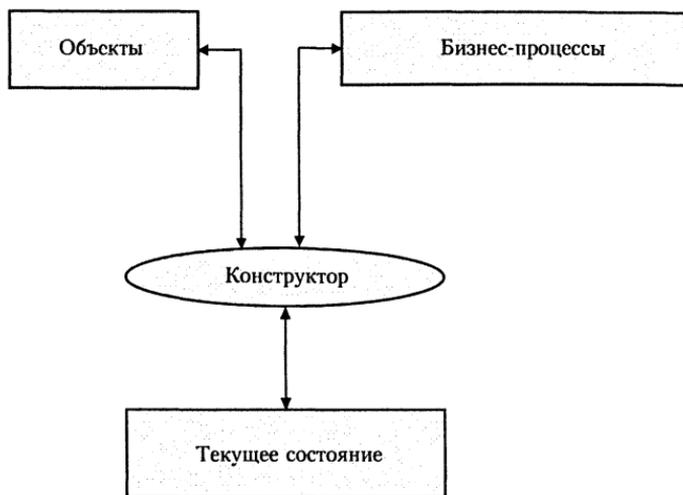


Рис.3.9. Компоненты среды проектирования

задача при всей ее сложности, в своей сущности, сугубо информационная. В машиностроении базисом для такого способа обучения выступает информатика и ВТ.

За период обучения конструктор досконально изучает важнейшие компоненты среды проектирования (рис.3.9). Поэтому переход выпускника в сферу высокой науки, производства или бизнеса осуществляется практически безболезненно, во всяком случае, без длительного и мучительного периода профессиональной адаптации. И уже для многих выпускников МГТУ «Станкин» термин «молодой специалист» не является синонимом «недоучки».

3.2. Возможности компьютерно-технологической среды в ходе промышленного проектирования изделий машиностроения

Применение ПЭВМ позволяет существенно изменить привычный облик традиционных профессий в машиностроении. Эти изменения связаны с возможностями:

- повышения качества конечных результатов;
- интеграции отдельных видов деятельности в рамках всего жизненного цикла изделия;
- накопления и применения повторно-используемых компьютерных баз знаний при проектировании и производстве наукоемких изделий;

- сокращения числа ошибок и при необходимости контролируемого их исправления в ходе практической деятельности;
- устранения утомляюще-однообразного характера исполняемой работы;
- предоставления большего времени для творческого труда;
- интенсификации интеллектуальной деятельности.

Как показывает практика и опыт работы в различных графических системах пользователю – машиностроителю, использующему компьютерную среду в своей повседневной деятельности, должны быть предоставлены такие условия работы, при которых он всегда может:

- оперировать понятиями и правилами, которые составляют суть его профессиональной деятельности и используются им в ходе выполнения повседневной работы;
- знает ЧТО и КАК ему можно и нужно делать в текущей ситуации дальше для выполнения производственной цели;
- решить текущую задачу, если не в полностью автоматическом режиме, то в автоматизированном режиме, используя ПЭВМ при выполнении отдельных доработок в ручном варианте;
- вернуться к ранее полученному решению и при необходимости может рассмотреть последовательность развития замысла об изделии в ходе его проектирования во времени.

С учетом перечисленных условий можно сформировать в обобщенном виде модель взаимосвязи компонентов технологической среды для реализации процесса проектирования в машиностроении (рис.3.10).

Жизненный цикл создания наукоемкого изделия можно описать как суперпозицию взаимодействий:

«СУБЪЕКТ» – I. Механизмы воздействия на изделие – «ИЗДЕЛИЕ» – II. Выработка представления о текущем состоянии изделия – «СУБЪЕКТ».

В компьютерно-технологической среде возможности воздействия пользователя во многом определяются восприятием текущего состояния «СУБЪЕКТА» об «ИЗДЕЛИИ». Это воздействие в последнее время основывается на использовании Windows – ориентированного интерфейса: окна, пиктограммы, меню, указатели и т.д. и обычно напрямую зависит от интеллекта, базовых знаний о жизненном цикле изделия и умения и навыков «СУБЪЕКТА» в работе с компьютерной средой. В обобщенной модели возможности «СУБЪЕКТА» по воздействию на «ИЗДЕЛИЯ» отражены в пункте А. В обобщенной модели условия выработки представления у «СУБЪЕКТА» о текущем

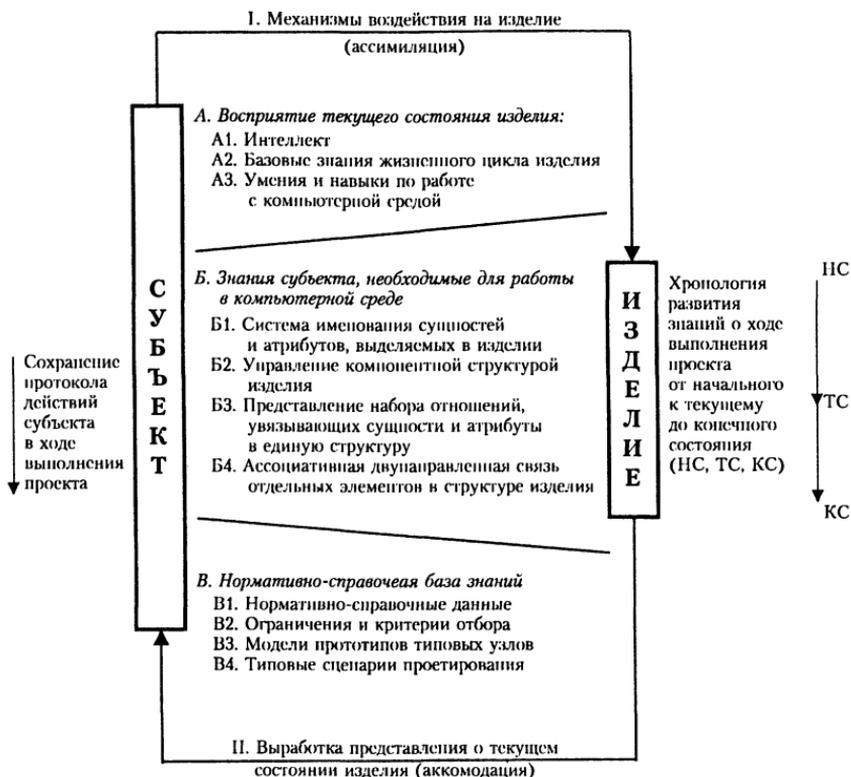


Рис.3.10. Обобщенная модель взаимосвязи компонентов технологической среды для реализации процесса проектирования в машиностроении

состоянии «ИЗДЕЛИЯ» перечислены в пункте В. В пункте Б обобщенной модели указаны те знания, которыми должен владеть «СУБЪЕКТ» при использовании компьютерной среды.

В компьютерной среде для удобства работы пользователя необходимо обеспечить сохранение протокола действий «СУБЪЕКТА» в ходе выполнения проекта. Эта необходимость связана с необходимостью сохранять существенные контрольные точки решения, в которые пользователь может вернуться при неблагоприятном развитии процесса проектирования в ходе дальнейшего выполнения работ.

Как показывает практика для пользователя очень удобно наличие средств ведения хронологии развития знаний о ходе выполнения проекта от начального состояния до конечного. Между этими крайними состояниями пользователь может иметь набор текущих состояний, который определяется только объемом наличной внешней памяти на компьютере пользователя.

Представленная модель в общем виде соответствует организации работ пользователей с различными компьютерными технологическими средами. В эту модель укладывается большинство современных графических систем, образующих основу для построения корпоративной информационной системы предприятия в машиностроении. В большей степени различие между отдельными графическими системами связано с тем, что понимают разработчики последних под термином «ИЗДЕЛИЕ» в обобщенной модели. В табл.8 приведены целевые установки, указывающие для реализации каких «ИЗДЕЛИЙ», по мнению разработчиков, предназначена та или иная компьютерно-технологическая система. Соответственно и содержательное наполнение пунктов А, Б и В обобщенной модели зависит от того, как трактуется термин «ИЗДЕЛИЕ» разработчиками компьютерно-технологической среды. В конце концов это наполнение выливается в комплект поставки и тот набор услуг, который предоставляет разработчик для пользователей при продаже той или иной компьютерно-технологической среды.

Таблица 8

Декларируемые цели создателей графических систем, используемых при решении задач конструкторско-технологической подготовки производства с помощью ЭВМ (на примере России в 2002 г.)

| Осознанность актуальности цели (в хронологическом порядке) | Массовость использования, в % (в скобках приведен накопленный процент) | Формулирование разработчиками основной цели создания графических систем | Типичный представитель: <i>иностранная</i> русская компания |
|--|--|--|--|
| 1985 | 35 | Формирование чертежно-конструкторской документации | <i>AutoDESK</i> Аскон |
| 1990–1995 | 20 (55) | Построение твердотельной модели | <i>UGSolutions (UGS)</i> Топ системы |
| 1990–2001 | 5–10 (60–70) | Исследование и учет поведенческих характеристик разрабатываемого изделия | <i>UGS,</i> <i>PTC Int.</i> Центр АПМ |
| 2000 | 20 (80–90) | Компьютерная разработка основного изделия и технологической оснастки для его производства | <i>UGS, IBM, SolidWorks,</i> <i>DEL CAM</i> ИКТИ РАН–СТАНКИН |
| 2003 | 15 (95–100) | Сохранение и использование накопленного опыта разработок в виде компьютерных баз знаний по подготовке производства | <i>UGS, IBM,</i> ИКТИ РАН–СТАНКИН |
| 2005 | 0–5 (100) | Работа в Internet среде | <i>UGS, PTC Int.</i> ИКТИ РАН |

Например, базовый программный продукт AutoCAD фирмы Autodesk прекрасно реализует в компьютерной среде наиболее востребованное в машиностроении представление «ИЗДЕЛИЯ» в виде комплекта чертежно-конструкторской документации. Экспертно потребность в использовании этого типа представления изделия на машиностроительных предприятиях в России в 2001 г. можно оценить в 35%. Все, что нужно для получения и работы с комплектом чертежно-конструкторской документации, в AutoCAD решено с блеском. Базовые знания жизненного цикла изделия определены и зафиксированы стандартами ИСО и ЕСКД. Если пользователь умеет работать с Windows, то умений и навыков по работе с компьютерной технологической средой ему обычно вполне достаточно. Используемая компьютерная база знаний обширна (в объеме трехтомного справочника Анурьева), но проста в построении, так как в основном предназначена для решения задач нормоконтроля.

В настоящее время в мире наиболее развитое представление «ИЗДЕЛИЯ» в машиностроении связано с компьютерной разработкой основного объекта и технологической оснастки для его производства. Массовость востребования этого типа представления изделия на машиностроительных предприятиях в России в 2001 г. экспертно можно оценить на уровне 20%. Правда реализация этого типа «ИЗДЕЛИЯ» предусматривает включение других форм представления в качестве основы для построения пирамиды. Так что в итоге массовость использования этого типа представления «ИЗДЕЛИЯ» может составлять 80–90%.

К наиболее типичным зарубежным представителям этого направления трактовки понятия «ИЗДЕЛИЕ» можно отнести, например, программные продукты UGS, CATIA, SolidWorks, PowerSOLUTION. Здесь следует отметить, что в большинстве случаев зарубежные программные продукты практически не адаптированы к условиям эксплуатации в России (по стапачному парку, по материалам, режимам эксплуатации, нормализованным компонентам и т.д.).

Этапы по совершенствованию деятельности машиностроительного предприятия с использованием компьютерных технологий приведены на рис.3.11. Как показывает практика, наличие перечисленных этапов требуется всегда. Простая покупка даже самой совершенной графической системы без выполнения перечисленных этапов не гарантирует непосредственный переход к совмещенному проектированию основного изделия и технологической подготовки его производства.



Рис.3.11. Этапы перехода от традиционной организации производства к компьютерной поддержке жизненного цикла изделия

3.3. Направления развития автоматизированных систем при выпуске наукоемкой продукции в единичном и мелкосерийном производстве

Анализируется роль и возможности систем автоматизированной поддержки информационных решений (САПИР) для машиностроительных предприятий. Основные цели, возникающие перед разработчиками этих автоматизированных систем, сводятся к формированию условий для повышения качества создаваемых технических систем и сокращению сроков их разработки и освоения в производстве с минимальным привлечением сторонних организаций.

Существующие сегодня на рынке системы автоматизированного проектирования (САПР) в машиностроении ориентированы преимущественно на конкретные проектные задачи и процедуры. В первую очередь это относится к проектным задачам и процедурам, связанным с геометрическим моделированием. В этих условиях потребители ожидают появления и распространения интегрированных инфраструктур для создания автоматизированных систем (АС) нового поколения, охватывающих весь комплекс работ по проектированию, моделированию и подготовке производства технических систем в компьютерной среде. Далее информационные системы нового поколения будем именовать как системы автоматизированной поддержки информационных решений (САПИР). САПИР призваны предоставить создателям технических систем (ТС) значительно более широкие возможности для повышения эффективности всего комплекса работ и в первую очередь проектных работ на ранних стадиях цикла проектирования при единичном и мелкосерийном характере производства. В результате применения АС нового поколения конструкторы ТС смогут выполнять более значительную часть проектных работ на абстрактных и понятийных уровнях описания. Это в свою очередь позволит повысить качество и свести к минимуму трудности реализации и отладки новых объектов и подготовки их производства.

Эволюция АС вызвана следующими моментами:

- ростом потребностей пользователей, а это в свою очередь ведет к росту сложности создаваемых технических систем,
- совершенствованием программно-технической платформы Wintel,
- относительным снижением стоимости профессионального рабочего места на базе ПЭВМ,
- наращиванием возможностей по компонентному программированию.

Практика показывает, что сложность создаваемых технических систем удваивается примерно каждые три года. Поэтому

стандартным инструментом пользователей промышленных предприятий вместо карандаша и кульмана по необходимости становится АС. По прогнозам специалистов в XXI в. более 80% всех новых разработок будет выполняться с помощью автоматизированных систем. Однако сами эти системы будут существенно отличаться от тех, что представлены на рынке сегодня.

В начале 1990-х годов средства автоматизированного проектирования на базе инженерных АРМ предназначались для удовлетворения вполне конкретных потребностей конструкторов технических систем: автоматизированного выполнения чертежно-конструкторских работ. Эти средства проектирования изначально были ориентированы на решение конкретных проектных задач, поэтому имели низкий уровень унификации. И, как следствие этого, каждое из них обычно имело свой собственный интерфейс, свою базу данных и свои условия эксплуатации. Техническую базу для реализации инженерных АРМ обычно составляли рабочие станции.

С ростом уровня автоматизации и эффективности решения отдельных проектных задач на предприятии наиболее узким местом стала организация взаимодействия и передачи данных между всеми проектными процедурами. Именно поэтому в последние годы на видное место выдвинулись работы по созданию корпоративных информационных систем, организующих и упорядочивающих движение проектной информации в рамках предприятия.

Критический рост уровня сложности создаваемых технических систем заставил разработчиков АС сконцентрировать свое внимание не только на интеграции всех производственных процессов в целом, но и потребовал перехода к использованию описаний проектов создаваемых ТС на более высоком уровне формализации. Эти описания составляют основу для последующего моделирования проектов и создания архивов типовых проектных решений в виде компьютерных баз знаний. Такое проектирование на высоком уровне описания стало возможным благодаря методам и средствам синтеза проектных решений. Последние средствами информационной технологии преобразуют высокоуровневые описания узлов и блоков в конкретные решения проектируемых ТС.

Два новых понятия – повторное использование знаний и совмещенное проектирование – являются ключевыми при интеграции всего процесса создания технических систем в рамках компьютерного проектирования и производства. Средства многократного использования инженерных знаний призваны обеспечивать хранение и вызов всей существующей и относящейся к делу проектной информации на данный момент времени. Это позволяет конструктору привлечь к решению конкретной задачи весь проектный опыт, накопленный им и его

предприятием. Совмещенное проектирование синхронизирует всю деятельность предприятия так, чтобы каждая стадия разработки выполнялась с учетом потребностей всего жизненного цикла создаваемой ТС.

При таком подходе к разработке наиболее существенные изменения относятся к ранним стадиям проектирования. Перемещение проектных ресурсов к началу цикла проектирования призвано снизить общий уровень затрат. Это связано с тем, что САПИР включаются в работу именно там, где влияние проектных ресурсов на свойства создаваемой технической системы максимально. Из практики известно, что основные технические решения, принимаемые на начальной стадии разработки, чрезвычайно сильно влияют на возможность своевременного выпуска конкурентоспособного объекта (рис.3.12). С точки зрения снижения стоимости единицы продукции и разработки в целом точное определение технических требований и разумное разбиение создаваемой технической системы на стандартные, типовые и оригинальные компоненты означают намного больше, чем эффективная реализация и изготовление самой продукции.

Поскольку на ранних стадиях проектирования серьезные ошибки связаны с относительно высоким риском неудачи всей разработки, повышением денежных и трудовых затрат, то оказание помощи конструктору на этих стадиях позволяет минимизировать как вероятность появления последующих ошибок, так и необходимость переработок проекта в будущем. Эта экономия резко сокращается по мере завершения цикла проектирования ТС и связано с тем, что ошибки, выявленные на более ранних стадиях создания, значительно легче исправить, чем ошибки, выявленные на более поздних стадиях.

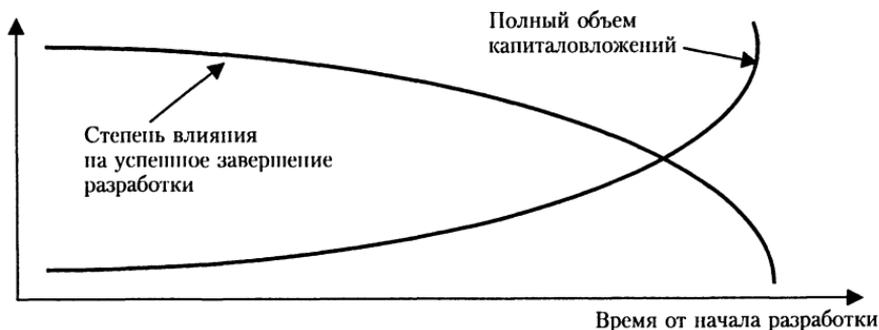


Рис.3.12. Влияние времени обнаружения ошибки на стоимость ее исправления

3.3.1. Составление технического задания на систему

Обобщенная модель деятельности КБ в ходе выполнения работ по КТПП приведена на рис.3.13.

Для создания высококачественного проекта технической системы необходимо учесть все функции, которые последней придется выполнять в течение всего срока службы. Другими словами, конструктор должен убедиться, что все требования, определяющие ход проектных работ, правильны и достаточны полны. Следовательно, роль САПИР должна измениться так, чтобы последние помогали концентрировать внимание конструкторов на том «ЧТО» он должен создать, а не на том «КАК» он это делает (рис.3.14). Обеспечивая более полную интеграцию всех аспектов, касающихся «КАК», вести проектирование, т.е. поднимая конструктора над проблемами ускорения каждой отдельной проектной процедуры и сосредоточив внимание на процессах проектирования в целом, САПИР призвана помочь



Рис.3.13. Обобщенная модель деятельности КБ в ходе создания конструкторско-технологической документации по контракту



Рис.3.14. Инфраструктура САПИР ориентирована на то, «ЧТО» должно быть создано, а не на то, «КАК» это надо сделать (на примере АС КТПП электродвигателя)

конструкторам переключить свой творческий познавательный потенциал на правильность определения будущих функциональных возможностей технической системы, т.е. именно на то «ЧТО» он создаст.

Техническое задание, полученное от заказчика, принимается за отправную точку разработки. При этом считается, что это техническое задание может подвергаться изменениям и развитию в ходе

уточнения, а может быть, и на более поздних стадиях. Эта деятельность выполняется совместно с заказчиком и заканчивается только при нахождении удовлетворяющих обе стороны компромиссов.

Сегодня процесс проектирования построен таким образом, что сразу же после формулирования технических требований начинается процесс реализации проекта, т.е. распределение отдельных частей этих требований между функциональными подразделениями проектной организации. При этом делаются попытки проверить и подтвердить правильность и согласованность этих требований путем проведения встреч и совещаний всех заинтересованных сторон, включая технических специалистов, специалистов по маркетингу и представителей заказчиков. Однако анализ и обсуждение этих требований часто приводит к тому, что на таких совещаниях нельзя исключить их неправильное толкование. Это связано с тем, что результаты обсуждения фиксируются на естественном языке, который допускает двусмысленности, недоговоренности, а иногда и заведомую ложь.

В АС нового поколения строится прототип компьютерной модели создаваемой технической системы, на котором конструктор и его заказчик смогут выполнять анализ и оценку технических требований в виде формализованного технического задания. В конечном итоге процедура проверки и подтверждения требований будет включать в себя сочетание моделирования с «оживлением» модели компьютерными средствами. Это позволит сделать технические требования чрезвычайно информативными и выразительными, т.е. по существу получить «виртуальный прототип» технической системы в виде компьютерной модели объекта.

Автоматизированная проверка технических требований состоит из двух частей:

- способа связного и полного описания этих требований;
- средства, позволяющего реализовать эти описания.

Данный метод позволит сформировать «компьютерную модель объекта», которая в формализованном виде содержит требования к ней. На этой стадии основу для моделирования составляет такое определение технической системы, в котором конкретные решения, связанные с применением расчетных, функциональных и механических блоков и узлов еще определены только в общем виде. Это обеспечивает конструктору полную свободу для описания и изучения функциональных потоков и компонентов технической системы для выбора рационального способа ее реализации.

Автоматизированная проверка технических требований по сравнению с ручной проверкой способна не только сократить число ошибок, но и призвана обеспечить более тщательное проведение данной работы. Это достигается на основе того, что экспериментировать с

компьютерной моделью значительно дешевле и проще, чем проводить испытания с натурными физическими моделями или макетами. В результате конструктор получит возможность намного более полно анализировать и учитывать особенности технических требований к проектируемой системе конкретного заказчика.

3.3.2. Разбиение систем на компоненты

После формулирования требований к технической системе конструкторы должны выделить в ней расчетные, функциональные и механические компоненты. Этот этап разработки обычно выполняется эмпирически на базе опыта и знаний специалистов прикладной области, поскольку создание моделей всевозможных комбинаций компонентов оказывается неприемлемо длительным. Однако со временем в состав АС будут включаться средства моделирования на высоких уровнях описания. Назначение такого моделирования призвано обеспечить численные сравнения различных результатов, полученных при использовании тех или иных вариантов разбиения технических систем. Этот этап разработки, который будем называть «прогноznым анализом», призван позволить конструктору избежать возможных неприятных сюрпризов при конкретной реализации технической системы.

По мере роста сложности технических систем анализ различных вариантов их покомпонентного разбиения становится все более важным. Однако одновременно усложняется и задача создания опытных и макетных образцов. Поэтому первая очевидная сфера применения такого прогнозного анализа – это те случаи, когда аппаратное макетирование невыполнимо или принципиально невозможно.

Используя «прогноznый анализ», конструктор получает возможность рассчитать результаты реализации системных функций в виде конкретных технических объектов и конкретных компонентов. Формируя исходные данные, необходимые для этих расчетов, САПИР будет активно взаимодействовать с существующими базами данных и прикладными средствами проектирования.

3.3.3. Средства для макетирования технических систем

Проведение прогнозного анализа потребует создания инструментальных средств, с помощью которых конструктор сможет оперативно строить макеты создаваемых технических систем. В качестве примера можно назвать систему моделирования условий штамповки, объединяющую в себе компьютер, программно-формируемые изображения условий штамповки, графические дисплеи и средства управления. Все вместе эти составляющие образуют для технолога программную среду для компьютерного моделирования технологии штамповки. С помощью такой среды обычно удается определить, что многие

проектные требования в технических заданиях пропущены или несовместимы друг с другом. В результате такие «виртуальные прототипы» будут полезны на всех уровнях процесса разработки – от формирования технических требований до практической реализации отдельных узлов и блоков.

Постепенный сдвиг в сторону подобных макетных систем наблюдается уже сегодня. В некотором смысле эти системы используются для интеграции всех функций разрабатываемой системы на максимально ранней стадии ее разработки. В основном такие средства уже присутствуют на рынке, однако они все же не обеспечивают возможности полного макетирования всех функциональных частей и узлов проектируемых технических систем в их конечном виде. Данная технология скорее будет применяться главным образом для выполнения многовариантного анализа типа «ЧТО БУДЕТ, ЕСЛИ ...»

Для решения подобных задач САПИР должна включать в себя всю информацию о технической системе, которую конструктор собирается создать. Например, для каждого из вариантов анализа типа «ЧТО БУДЕТ, ЕСЛИ ...» конструктор должен иметь в своем распоряжении соответствующую технологическую информацию, чтобы производителям не пришлось в дальнейшем при освоении объекта в производстве дорабатывать и корректировать его проект. Исторически сложилось такая ситуация, что различные средства автоматизированного проектирования «понимали и воспринимали» лишь небольшие части полного проекта. Например, программа моделирования понимала один из видов описания проекта технической системы, а средства автоматизированного конструирования – совсем другой вид описания.

Идеальный вариант разработки состоит в том, чтобы перед запуском любого технического объекта в производство можно было бы обойтись всего одним вариантом его натурального образца. Желательно, чтобы все остальные проблемы «интеграции» отдельных автоматизированных системы можно было решить с помощью единого информационного пространства для решения всего комплекса производственных задач. Следовательно, конструктор должен быть уверен, что с помощью имеющихся у него средств проектирования он может создать и ввести полное описание своего объекта в предполагаемых условиях его функционирования, т.е. по существу предварительно «построить» этот объект в некоторой «виртуальной лаборатории». Для приближенного расчета характеристик создаваемой технической системы эти средства, возможно, будут работать на основе экстраполяции существующих данных. Однако при этом все параметры и свойства создаваемой технической системы должны быть оценены как можно полнее и точнее.

Поскольку отдельные технические требования в ходе разбиения проектируемого объекта на компоненты и собственно проектирования могут меняться, чрезвычайно важно на каждой стадии обеспечивать «трассируемость» (связность) требований и конкретных технических решений. Поэтому в САПИР предусматриваются средства и функции идентификации тех конкретных требований, которые вызвали потребность создания каждого элемента проекта. Чтобы облегчить модификацию проекта, эти средства трассировки будут также идентифицировать те части высокоуровневого описания проекта, в которых фигурирует этот элемент, и те фрагменты низкоуровневого описания, из которых этот элемент конкретно состоит.

Такая «трассируемость» позволит конструкторам эффективно работать с разными вариантами представления проекта, с которыми ему приходится иметь дело при разработке любой технической системы. Если при описании проекта на различных уровнях иерархии обнаруживаются противоречия, то САПИР должна позволить конструктору их обнаружить и устранить.

3.3.4. Использование проектного опыта

Одно из крупных изменений в технологии проектирования, связанное с применением информационных технологий, – это намного более высокий уровень повторного использования прежних проектных решений. Это значительно облегчает проведение проектных работ на высоком уровне абстракции. С помощью такого повторного использования – построение новых проектов на языке существующих проектных решений – конструкторы ТС получают возможность эффективного применения существующих данных с обеспечением совместимости и преемственности при высоких экономических показателях. Например, новая ТС обычно строится на основе предыдущей модели и содержит не более 10–20% действительно оригинальных технических решений. Современные средства АС мало приспособлены для эффективного и полномасштабного использования уже существующих проектных решений.

Повторное использование проектных решений позволяет эффективно управлять сложными создаваемыми техническими системами, применяя при проектировании ранее отработанные функциональные блоки. Такой подход дает возможность выполнять проекты с меньшим количеством более крупных функциональных узлов и блоков. Вводя нужные уровни абстракции и иерархии, конструктор ТС получает возможность с помощью паличных средств проектирования создавать гигантские по сложности проекты.

Чтобы практически реализовать повторное использование проектных решений, обычные проектные библиотеки необходимо

развивать в компьютерные базы знаний, которые предусматривают доступ к коллективной проектной базе знаний всего предприятия. Эти базы знаний будут содержать значительно более сложную информацию по сравнению с сегодняшними библиотеками – в частности, в них большое внимание будет уделено понятийной модели, кодированию «проектной истории», а не просто хранению отдельных проектных элементов. В этих базах знаний должны храниться не только описания заведомо работоспособных проверенных компонентов, но и целые фрагменты прежних проектов, работоспособность которых заведомо известна. В идеальном варианте конструктор сможет даже сохранять в них в том или ином виде условия применения, заложенных в них проектов.

В структуре этих баз знаний можно будет отображать иерархию выполняемых проектов, для чего хранимые описания проектов должны содержать ссылки на фрагменты, хранящие описания их компонентов на более низком уровне представления (рис.3.15). Такая иерархическая структура позволяет сохранить объем компьютерных баз знаний на приемлемом с точки зрения управляемости уровне и обеспечить эффективную организацию перекрестных ссылок для проектных альтернатив.

Чтобы организовать хранение всей этой дополнительной информации, а не только конечных результатов отдельных процессов проектирования, в автоматизированной системе проектирования потребуются специальные средства наблюдения. Эти средства будут регистрировать последовательность проектных операций в виде сценариев проектирования в успешно завершенных проектах. Последние в

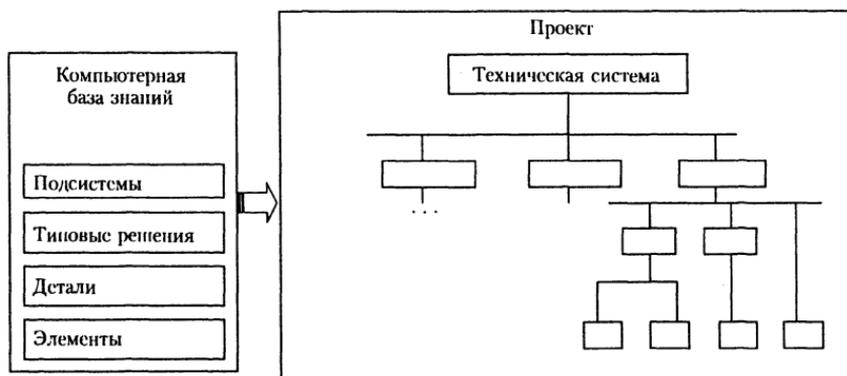


Рис.3.15. Компьютерные базы знаний и их использование в проектах при реализации технических систем

дальнейшем могут послужить в качестве руководящих принципов при выполнении будущих проектов. Кроме того, эти средства можно использовать для передачи успешных проектных решений в средства синтеза и тем самым расширять базу знаний последних.

Описанная компьютерная база знаний позволит не только отслеживать взаимосвязь компонентов и проектов, но и обеспечит определенное состояние их хранения с точки зрения полноты информации о функциональных возможностях, тестируемости, надежности, технологичности и удобстве сопровождения в эксплуатации. Для вновь вводимых в компьютерную базу знаний компонентов или фрагментов специальный персонал будет получать автоматические сообщения о необходимости выполнить их подробный анализ по всем указанным показателям, что значительно облегчит деятельность предприятия по обеспечению высокого качества продукции.

3.3.5. Повышение качества программ поиска информации

Для практической работы с большими компьютерными базами знаний в АС потребуются сложные программные средства поиска и фильтрации информации. Эти программы должны выполнять «разумный» поиск на тех стадиях, когда полный набор технических требований еще не сформирован. Например, конструктор должен иметь возможность сформулировать запрос в виде «Мне нужен аналог примерно такого типа электродвигателя» и получить ответ на него в виде набора конструкций, максимально соответствующих предъявленным требованиям.

Когда в автоматизированной системе проектирования предусмотрены эффективные средства повторного использования проектных решений, то сокращенные по сравнению с сегодняшней численностью проектные группы могут быстро разрабатывать надежные системы на основе существующих проектов. При этом большинство решений уже проверены, и все характеристики известны, в тоже время для каждого из решений не потребуется привлекать широкий круг технических экспертов для анализа совокупности показателей проекта. Небольшие проектные группы смогут успешно работать над созданием высокопроизводительных и сложных технических систем, которые потребуется разрабатывать в обозримом будущем.

Другая технология проектирования, способная помочь конструкторам перейти на более высокие уровни абстракции при проектировании, – это технология синтеза проектных решений, автоматизирующая их реализацию и вместе с тем позволяющая специалистам уделять больше времени начальным стадиям системного проектирования. Такой переход по своему содержанию близок к внедрению алгоритмических языков высокого уровня в деятельности инженеров-программистов.

Именно по такому сценарию сегодня идет и развитие технологии проектирования технических систем. Программы логического синтеза и генераторы модулей позволяют конструкторам технических систем составить описание некоторого компонента или подсистемы, которое затем в полностью автоматическом режиме обрабатывается системой проектирования «до конца». Как в свое время компиляторы с алгоритмических языков, средства синтеза проектных решений выходят сегодня на такой уровень эффективности, что конструктора «доверяют им» создание фрагментов, осваиваемых затем в производстве.

Работа на языке более высокого уровня к тому же «приближает» соответствующий язык программирования к решаемой с его помощью прикладной задаче. Специалисту-прикладнику требуется такой язык, который имел бы смысловое содержание в соответствующей предметной области. Причина состоит в том, что гораздо эффективнее писать программу для решения некоторой задачи специалисту-прикладнику, который знает эту задачу, чем профессиональному программисту, знающему машинный язык.

Применение средств синтеза облегчает организацию процессов нисходящего последовательного проектирования. Однако возможна такая ситуация, когда в процессе реализации проекта желательно внести в него какие-либо изменения. Возникает вопрос: как определить влияние этих изменений на верхние уровни иерархии описания проекта или на проект в целом?

В этой ситуации можно вернуться на верхний уровень описания проекта и повторить задачу синтеза с новым набором проектных ограничений, получив новый вариант ее реализации. Такая процедура позволит обеспечить непротиворечивость всех уровней описания проекта, но при этом конструктор должен быть уверен, что он имеет возможность управлять процессом синтеза в направлении той оптимальной реализации некоторого фрагмента, которую он хочет получить. К сожалению, современные методы и средства синтеза проектных решений такой гарантии ему не дают.

На практике обычно возникает задача модификации тех синтезированных вариантов, которые не удовлетворяют всему набору технических требований к создаваемой системе. После этого необходимо обеспечить непротиворечивость и совместимость измененной конкретной реализации проекта и его высокоуровневых описаний. Эту непротиворечивость можно проверить и при необходимости устранить вручную или воспользоваться средствами смешанного моделирования. При этом конструктор сможет сравнить новый вариант реализации технической системы с его предполагаемыми функциональными характеристиками.

3.3.6. Обратный инжиниринг

Но есть и другой вариант – воспользоваться промежуточной проектной технологией, которую определим как обратный синтез. Эта технология позволяет изменить фрагменты высокоуровневого описания проектируемой системы в соответствии с изменениями, внесенными на более низких уровнях описания. Метод обратного синтеза переносит изменения в некотором элементе более низкого уровня описания во все фрагменты описания более высокого уровня, в которые этот элемент входит.

При всей своей полезности и удобствах средства синтеза на основе накопленных компьютерных баз знаний и средства повторного использования проектных решений не обеспечивают полной автоматизации процесса проектирования. Например, конструктору может потребоваться выполнить численную оценку влияния выбора того или иного технического решения. В этом случае он может синтезировать соответствующую реализацию проекта или подобрать существующий вариант, близкий к необходимому, и провести его подробное исследование.

При таком подходе средства синтеза и повторного использования проектных решений действительно помогут разработчику найти наилучшую комбинацию компонентов проекта. Главное преимущество этого подхода состоит в том, что конструктор получает возможность «зондировать» важнейшие точки внутри создаваемой системы.

Средства повторного использования проектных решений и средства синтеза на деле дополняют друг друга. Для реализации до 90% технической системы, если проектирование ведется на основе аналога, требуются первые из этих средств. Затем конструктор проводит экспериментальные исследования новых частей технической системы, используя средства синтеза для работы по методу «что будет, если...», вплоть до выхода на самый нижний уровень описания проекта.

Внедрение средств повторного использования проектных решений и синтеза не только позволяет сократить численность проектных коллективов, но и существенно изменить требования к составу специалистов. В результате использования САПИР эти группы начнут комплектовать специалистами разного профиля, поскольку при новых подходах к проектированию все аспекты разработки технической системы можно и нужно будет прорабатывать параллельно. Только в этом случае проектная группа окажется в состоянии полностью охватить всю совокупность показателей и характеристик проекта и принять квалифицированные технические решения.

Данный подход к разработке технических систем называется совмещенным проектированием. Правилom при комплектовании проектных коллективов станет включение в их состав специалистов по расчетам, конструированию, технологической подготовке производства,

которые будут одновременно обрабатывать все аспекты создаваемой технической системы. Средства проектирования, используемые каждым специалистом группы, будут иметь доступ к общим базам знаний и данных. Для полного охвата всего круга вопросов создания технических систем в такие проектные коллективы войдут также специалисты по надежности, испытаниям, эксплуатационному обслуживанию и т.д.

Такое сотрудничество специалистов из разных областей техники дает многочисленные преимущества. Комплексный подход с объединением разных специалистов в одном коллективе облегчает контроль и управление такими «внешними» вопросами, как ресурсы, технические требования и поставщики. Все эти факторы в ходе разработки и производства объекта могут изменяться.

При работе по методу совмещенного проектирования конструктор может рассчитывать, что конечные характеристики проектируемой системы станут ему известны на самых ранних стадиях разработки, и он может сравнить эти характеристики с требованиями и ограничениями технического задания. Но чтобы иметь действительно содержательные прогнозные данные, конструктору потребуются все перечисленные выше средства проектирования: средства анализа проектов на высоких уровнях описания, развитые компьютерные базы знаний, средства синтеза проектных решений и общая база данных.

По оценкам экспертов использование совмещенного проектирования в ходе оптимизации изделия и технологического процесса его изготовления позволило снизить:

- объем заказов на конструктивные изменения более чем на 50%;
- продолжительность разработки изделия – на 60%;
- количество неудачных проектных решений и объем доработок – на 75%.

САПИР помогают конструкторам быстро создавать высоконадежные технические системы высокого уровня сложности при мелкосерийном и единичном характере производства. Средства проектирования и моделирования на высоком уровне описания позволяют подробно исследовать создаваемую техническую систему и оценить ее параметры и характеристики еще до начала конкретной деятельности и до начала реализации собственно проектных решений. По мере движения по проектной иерархии сверху вниз конструкторы смогут в полной мере использовать накопленные проектные решения и средства синтеза новых решений. А метод совмещенного проектирования гарантирует, что при реализации технической системы будут в полной мере учтены все требования и ограничения к ней.

В результате конструкторы смогут больше времени и сил отдавать проектированию на высоком уровне описания, будучи

уверенными, что последующая реализация технических решений пройдет у них быстро и без ошибок.

Что же дает применение информационных технологий?

- Резкое снижение затрат времени, числа работников и других затрат на выполнение производственных функций.
- Снабжение работника всеми возможностями для самостоятельного получения конечного результата.
- Возможность работы на будущие потребности клиента.
- Ускоренное продвижение новых технологий.

3.4. Особенности анализа и синтеза процессов в машиностроении при переходе к автоматизированным системам

3.4.1. Два подхода к технологии создания программного продукта

Традиционный подход к созданию программного продукта обычно проходит по следующей схеме (рис.3.16).

1. Конечный пользователь формирует свои требования к будущей программе на языке требований (L1). В них он указывает, для чего нужна программа, ее общие характеристики, свойства, платформу, на которой она будет выполняться и т.д.

2. Один из специалистов предметной области, выступающий в качестве заказчика, на основе требований отдельных пользователей и своих знаний создает спецификацию на языке спецификаций (L2). Здесь он в виде формул и других формализованных зависимостей описывает структуру данных решаемой задачи и формирует сценарий действий пользователя при решении поставленной задачи.

3. Разработчик-аналитик на основе спецификации заказчика составляет алгоритм решения задачи на языке проектирования (L3). Аналитик пытается построить алгоритм решения задачи таким образом, чтобы, с одной стороны, полностью решить задачу, а с другой стороны, сократить время и количество ресурсов ЭВМ, затрачиваемое на решение задачи. Алгоритм чаще всего представляется на каком-нибудь формальном языке (частный случай – язык структурных схем).

4. Разработчик-программист реализует алгоритм решения задачи на языке программирования (L4 – чаще всего универсальный язык высокого уровня). После этого обычно средствами стандартной системы программирования производится синтаксическая и семантическая отладка. Средствами стандартной системы программирования формируется загрузочный модуль программы (L6). Здесь следует

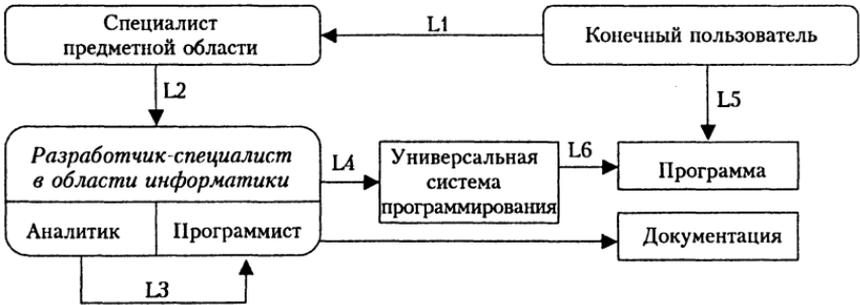


Рис.3.16. Компоненты традиционного процесса создания программного продукта.

L1 – язык требований; L2 – язык спецификаций; L3 – язык проектирования;
 L4 – универсальный язык программирования; L5 – язык общения;
 L6 – язык машинных кодов

отметить, что программа и документация на нее являются порождением различных процессов.

5. Конечный пользователь общается с полученной программой на языке общения (L5).

В этой схеме очевидны следующие недостатки.

1. Оторванность специалиста предметной области от процесса реализации программы. Это связано с тем, что, обладая профессиональными знаниями, специалист предметной области обычно не имеет достаточных знаний о способах использования вычислительных средств.

2. Наличие большого числа специалистов, участвующих в разработке; следствием этого обстоятельства является:

- появление эффекта «испорченного телефона», при котором пользователь после долгого и дорогостоящего процесса разработки обычно получает совсем не то, что ему нужно;
- длительность разработки;
- дороговизна создания программного продукта.

Вследствие жесткости связей между специалистами возникают сложности при отладке и сопровождении программы. После построения схемы решения обычно происходит полное отчуждение специалистов предметной области от процесса решения задачи. Если пользователь захочет, чтобы программа решала более широкий круг задач, или если он обнаружит в программе какие-либо ошибки, недочеты, неточности, то будет вынужден запустить весь процесс создания программного продукта вновь по полной программе.

Традиционный подход несет в себе не только указанные выше недостатки, но и недостатки традиционных средств создания программ, а именно

1. Алгоритмичность программы, так как основное внимание в традиционных системах программирования уделяется не описанию предметной области, а написанию правильного алгоритма решения задачи, что приводит к следующим ограничениям:

- жесткому делению переменных на входные и выходные;
- решению задачи, которое представляется в виде строгой последовательности процедур;
- отсутствию модульности при построении системы;
- невозможности работы программы с неполностью определенными данными.

2. Сложность взаимодействия с другими программами.

3. Документация на программу и собственно программа являются абсолютно разными объектами.

Традиционная технология программирования ориентирована на решение в основном формализованных задач. Большая же часть задач при компьютерной подготовке производства в машиностроении относится к классу слабо формализованных. Эту ситуацию можно преодолеть только путем привлечения специалистов предметной области к процессу разработки и сопровождения прикладного программного обеспечения.

Рассмотрим схему создания программного обеспечения с использованием ИКС (рис.3.17). Для этой схемы характерным является следующее.

1. Конечный пользователь формирует свои требования к будущей программе на языке требований (L1).

2. Специалист предметной области формирует спецификацию на программный продукт, описывая свои знания на специальном формализованном языке спецификаций (L2F). При этом специалист использует максимально привычные для себя формы представления знаний (формулы, таблицы), сложившиеся в машиностроении.

3. Система ИКС на основе подготовленной пользователем спецификации автоматически генерирует программу, полностью исключая процесс написания и отладки программ в традиционном смысле. Созданные спецификации понятны для пользователей и могут служить частью документации на созданную программу.

4. Конечный пользователь общается с полученной программой на языке взаимодействия (L5F). Особенности этого языка приведены в руководстве по ИнИС.

Использование такой схемы дает преимущества.

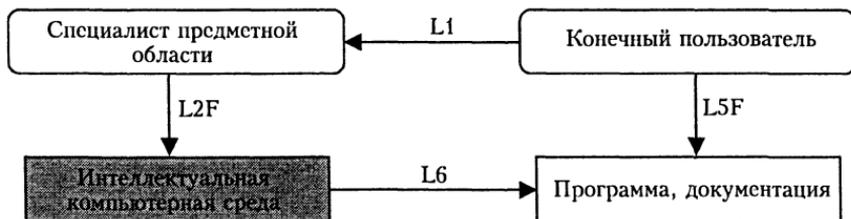


Рис.3.17. Компоненты процесса создания программного продукта с помощью ИКС.

L1 – язык требований; L2F – язык спецификаций ИКС; L5F – язык взаимодействия пользователя с программой; L6 – язык машинных кодов

1. Специалист предметной области ориентируется в первую очередь на создание модели объектов реального мира. В машиностроении деление на объекты и действия с ними достаточно хорошо отработаны в нормативно-справочной документации, что дает следующие преимущества:

- деление на объекты обеспечивает модульность системы;
- переменные являются характеристиками объектов предметной области;
- нет жесткого деления переменных на входные и выходные.

2. Исключение из процесса разработки программного обеспечения профессиональных программистов позволяет:

- уменьшить время разработки программного продукта в 10–15 раз;
- уменьшить число ошибок и сократить время отладки;
- упростить сопровождение и модификацию программы.

3. Исходная спецификация является одной из основных составляющих документации на программу.

3.4.2. Источники формирования требований к ПО

Известно, что качество автоматизированных систем (АС) не представляется в виде «коробочных решений», оно может быть только результатом осознанной, квалифицированной и достаточно длительной деятельности. Причем итоговый результат этой деятельности чаще всего определяется не где-то на «фабрике по производству программного обеспечения», а индивидуально на каждом конкретном предприятии.

Любая разработка, в том числе и программного обеспечения для предприятия в целом, начинается с формирования требований. Собственно требования к программно-информационной системе

предприятия формируются из целей бизнеса, определяемых руководством, существующих технических процессов, накопленного опыта, навыков и т.д. (рис.3.18). Использование на рис.3.18 овалов, выходящих за требования к программно-информационному обеспечению предприятия, может быть интерпретировано следующим образом. В случае 2 это та плата предприятия за покупку стандартного (чаще всего «коробочного») программного обеспечения, которое вряд ли когда-нибудь в полном объеме сможет применяться на конкретном рабочем месте. Случай 3 указывает на то, что абсолютно все технические процессы предприятия вряд ли когда-либо будут автоматизированы в полном объеме. Как показывает практика, цена каждого последующего процента при приближении к 100 % автоматизации резко возрастает. Случай 4 отражает некоторые психологические факторы, связанные с тем, что с накопленным программно-информационным багажом возможно частично придется расстаться, т.е. этим отражается ситуация «когда хранить уже накладно и выбросить жалко».

Существенное влияние на эффективность деятельности предприятий современных автоматизированных систем КТПП и их высокая стоимость вынуждают руководство последних вкладывать немалые финансовые ресурсы в информационные технологии. Такой *интерес к информационным технологиям*, в первую очередь, *связан с тем, что они могут предоставить гарантии и прибавить уверенности в получении качественного конечного результата*. Поэтому понятно желание руководства предприятия найти информационные технологии, универсально решающие все проблемы построения автоматизированных систем, обеспечивающих деятельность всего предприятия в целом. К сожалению, как показывает практика, такие технологии пока еще далеки от массового тиражирования. Поэтому чаще всего

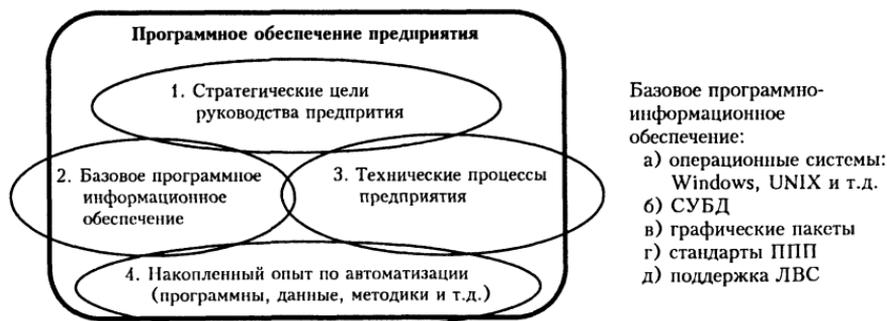


Рис.3.18. Источники формирования требований к программному обеспечению предприятия

существующие технологии и поддерживающие их инструментальные средства имеют свою область применения, свои недостатки и ограничения.

Как известно, когда у мастера основной инструмент молоток, то каждая проблема становится похожей на гвоздь. Подобная ситуация возникает и в случае освоения новой информационной технологии или приобретения дорогостоящей CASE-системы. Хочется все проблемы решать только теми методами, которые присущи этим средствам. Часто результатом такой деятельности становится разочарование в CASE-системах в целом. Внедрение информационных технологий откладываются до лучших времен, дорогие инструменты не используются.

Однако избежать внедрения новых информационных технологий на практике не удастся. Чем крупнее и сложнее должна быть создана автоматизированная система, тем выше риск ее неудачного построения. Одним из способов снижения риска, повышения гарантии выполнения в срок и при заданном финансировании построения АС является формализация процесса анализа и проектирования последнего. Такой подход помогает добиться повторяемости успеха и качества при информатизации того или иного вида деятельности. Излишне говорить о важности такого результата для предприятий, деятельность которых существенно зависит от функционирования автоматизированных систем, обеспечивающих проектирование и изготовление наукоемкой продукции. А чтобы освоение новых информационных технологий проходило эффективно, необходимо четко видеть их преимущества и недостатки.

3.4.3. Причины появления искажения требований к АС

Далее до конца главы использованы материалы из «Computer Week» и Интернет. Почти в каждом проекте возникают ситуации следующего типа. Заказчик что-то считает для себя очевидным и поэтому даже не упоминает об этом. Разработчик так же молча «обходит» часть требований как абсурдные или же «неудобные». В ходе проектирования сплошь и рядом одни защищают какие-то требования, другие говорят о них как о невыполнимых или безграмотных. Что и как надо сделать, чтобы противоречивые точки зрения, упущения и умолчания не исказили создаваемую АС? Как организовать выполнение работы по созданию АС, чтобы в результате последняя не была отторгнута потребителем или бесконечно долго не «доводилась до ума» под встречные упреки заказчика и исполнителя?

Одни говорят, что все проблемы решаются через создание прототипа автоматизированной системы, реализуемой непосредственно исполнителями с привлечением конечных пользователей заказчика.

Другие уверены, что дело «компьютерщиков» (включая программистов, сетевиков, проектировщиков баз данных и других специалистов) – сразу предоставить работающую автоматизированную систему в ответ на задание, которое выскажет заказчик. Первая точка зрения очень распространена среди исполнителей. Вторая свойственна заказчикам, относящимся к разработчику, как капризный клиент к портному: «Я плачу деньги, а ты сделай, чтобы мне было хорошо. И без лишних вопросов и примерок». Но и то и другое ведет к искажениям требований.

Вот почему в последнее время при создании больших автоматизированных систем появилась новая группа специалистов-аналитиков и их многочисленных «разновидностей». Ответу на вопросы ПОЧЕМУ и КАК они должны сегодня выполнять ключевую роль в построении АС и посвящена настоящая часть работы.

Корни проблемы в том, *что каждый отдельный человек, связанный с разработкой или использованием больших автоматизированных систем, обычно имеет не полное, а подчас и искаженное представление о всей картине создания и жизни последней*. Чисто по человечески это нормально и неизбежно. Попробуем разобраться в этом вопросе, представив взаимодействие пары «заказчик – исполнитель» с помощью модели Тыгу.

Источники искажений, возникающих при создании больших автоматизированных систем, показал Э.Х.Тыгу на предельно простой, но очень полезной модели. В идеальном варианте модели показаны ситуации, в которых возникают искажения восприятия основных участников проекта создания АС: «заказчика» и «исполнителя» (рис.3.19). На взгляд каждого из них проблемы другой стороны видны «издалека». При этом чужое становится мелким, а отчетливо видны детали только своих проблем.

Типичная ситуация искажений с точки зрения исполнителя изображена на рис.3.20. В этом варианте сознательно или несознательно исполнитель так ограничивает угол своего «наружного» взгляда, что из всех ситуаций заказчика видит только ту часть, которую легко воспринимает и для чего самому хочется написать программу. Немудрено, что получающееся в результате «изделие» или отвергается заказчиком, или заставляет испытывать долгие мучения в ходе своего внедрения.

Из приведенных моделей видно, что задача выработки общего подхода к созданию АС по разному стоит перед заказчиком и исполнителем. Особенно остро она проявляется в подходах к организации корпоративной разработки и управления изменениями АС при реализации проекта совместными усилиями.

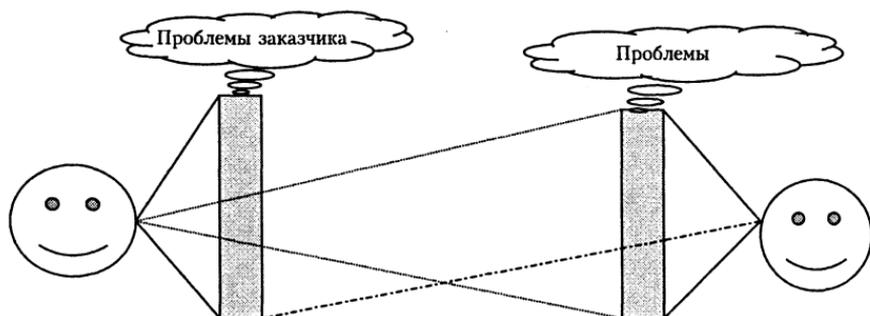


Рис.3.19. Взгляды заказчика и исполнителя на «свои и чужие» проблемы

Такой взгляд на разработку АС во многом определяется различием корпоративных культур заказчика и исполнителя. Обычно организация разработки АС исполнителем технологически и бизнес ориентирована, что фиксируется в виде:

- договора с заказчиком и технического задания;
- условий поставки АС и четко обозначенного набора услуг.

С этой точки зрения исполнителя практически мало интересует достижение конечного результата, ожидаемого заказчиком после внедрения АС на своем предприятии. Связанный с этим типичный пример работы исполнителя можно продемонстрировать на отношении к программному инструментарию, с помощью которого осуществляется автоматизация деятельности предприятия и его обслуживание. Исполнителя обычно мало интересует сколь эффективно будет работать этот инструмент в руках заказчика. В этом кроется одно из главных расхождений интересов заказчика и исполнителя при заказном

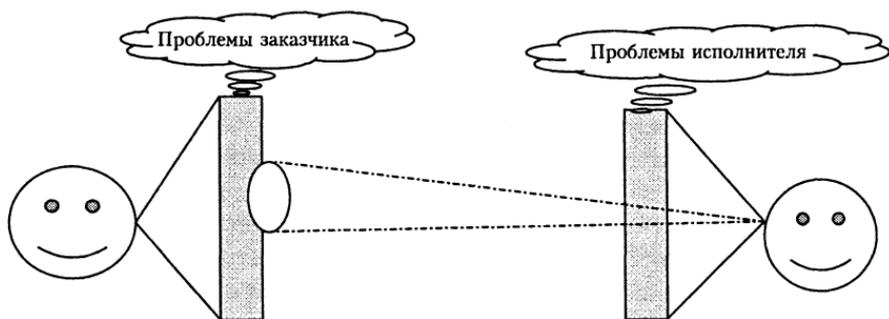


Рис.3.20. Асимметричный вариант модели Тьюгу

создании АС: *заказчику нужна эффективная и сегодня работающая АС, а исполнителю – реализация АС, не противоречащая условиям договора и технического задания.*

Из практики известно, что ошибки в реализации АС обходятся очень дорого. Архитектурные просчеты вообще могут привести к провалу проекта по созданию и внедрению АС на предприятии. Причем ошибки эти чаще всего всплывают только при попытке внесения изменений в созданную АС. Обычно ситуация достигает апогея в следующих случаях:

- технологии разработки программного обеспечения заказчика и исполнителя разные;
- заказчик обладает оригинальным накопленным функционалом по решению конкретных технических процессов;
- заказчик собственными силами осуществляет сопровождение и модификацию АС, созданную исполнителем.

Вот почему в качестве аксиомы при создании заказного ПО выступает следующий факт: *технология реализации АС должна быть максимально общей и для исполнителя и для заказчика.* В этом случае указанные критические ситуации удастся избежать, то хотя бы предотвратить.

Продемонстрировать какие именно специалисты и за счет чего могут помочь в решении проблемы согласования деятельности по разработке АС можно на основе «расширенной модели» (рис.3.21). Для этой модели к паре «заказчик – исполнитель» добавим двух новых участников: аналитика по техническим процессам и архитектора АС. В рамках расширенной модели попробуем показать их функции, профессиональные отличия друг от друга и от остальных участников, наметить связи между ними.

3.4.4. Аналитик по техническим процессам

Главные задачи аналитика по техническим процессам – правильно понимать потребности отдельных пользователей и предприятия в целом, ясно и полно их описывать в уместных формах. Аналитик ближе других находится к заказчику и к пониманию его технических процессов. Он «живет» в них. Принципиально, что он умеет обобщать, строить, документировать формальные и неформальные модели их решения. Он должен представлять себе общие возможности современных информационных технологий, в то же время не забывая себе голову вопросами, характерными для специалистов по компьютерным технологиям.

Обычно перед аналитиком по техническим процессам стоит дилемма участия в двух тесно взаимосвязанных пластах деятельности предприятия: технологическом и социально-руководящем. В

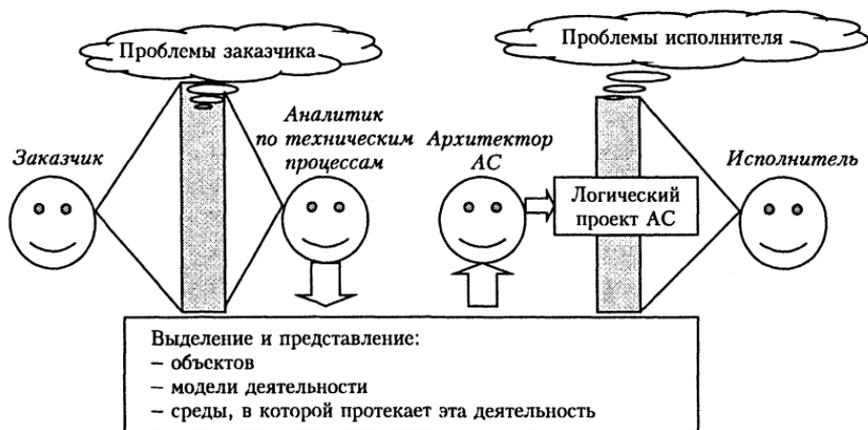


Рис.3.21. Расширенный вид модели Тытуг

последнее время эта дилемма резко обозначилась. Далекое не каждому специалисту предприятия приходится моделировать деятельность последнего, а тем более создавать ее действующую технологическую модель. Аналитику по техническим процессам неизбежно приходится сталкиваться и с тем и с другим. Именно с этим связан рост статуса на предприятии аналитика по техническим процессам.

Очень часто аналитик не является сотрудником фирмы – разработчика или специализированного подразделения предприятия – заказчика (например, отдела программирования). Современные методики предполагают, что даже консалтинговая фирма, имеющая в своем штате профессиональных аналитиков, ищет таких людей у заказчика, находит с ними общий язык и включает их в свой проектный коллектив.

От аналитика не требуется обязательного освоения формальных языков и моделей (таких как модель потоков данных или UML – «унифицированный язык моделирования»). Он может использовать формализмы, но только для более точного описания внешних требований к АС, а не для принятия решений в проектировании «компьютерной» части системы. Аналитик может и часто должен формировать не только постановки отдельных задач, но и целостную модель автоматизированного предприятия. Однако он строит такую модель на верхних уровнях архитектурного представления автоматизированной системы, в то время как CASE-модели лежат ниже.

Аналитику нужен очень простой и гибкий компьютерный инструмент для ведения документации: текстов, коллекций рисунков и схем, чертежей, записей алгоритмов и ссылок на нормативные документы с извлечением из них. Значительную часть своих наработок аналитику следует помещать в общую базу проекта, поддерживающую «требования к автоматизированной системе».

3.4.5. Архитектор АС

Архитектор рассматривает всю совокупность внешних требований к автоматизированной системе. Вместе с аналитиком он анализирует и выявляет противоречия и неточности в них. Он находится немного дальше от заказчика, чем аналитик, но ближе к компьютерным специалистам. С участием экспертов по компьютерным технологиям он проверяет требования на приемлемость. В случае слишком большой стоимости или невозможности реализации он ищет альтернативные подходы.

Архитектор создает целостный «логический» проект АС, в котором еще не производится выбор конкретных компьютеров, СУБД, инструментов программирования и т.д. Главное здесь то, что он видит и описывает архитектуру АС в целом, объединяя все ее «слои» — от людей и целей до данных и функций.

Архитектор принимает решения, интегрируя концептуальную модель предприятия с потенциально доступными возможностями компьютерных технологий. Ему нужно рассматривать и чисто технические решения, и часто вместе с проектировщиками автоматизированных систем приходится проверять выполнимость тех или иных требований и логических решений. Но окончательные решения по технической архитектуре он не принимает, это дело «главного программиста» АС, хотя многие такие решения должны согласовываться с архитектором.

Архитектор — сотрудник специализированного «компьютерного» подразделения или фирмы. Он владеет формальными моделями. Он обязан оценивать эффективность своих решений и присущий им риск. Главное здесь то, что он представляет коллектив разработчиков АС перед заказчиком. Он проверяет решения вместе с аналитиками и пользователями, он объясняет возможности АС первым лицам заказчика, защищает ее на совещаниях.

Иногда архитектор — это «главный архитектор АС», иногда — член «архитектурной мастерской». Но когда его функции выстроены правильно, тогда и заказчик и программист должны считать его не страшилой-начальником, а конструктором совершенно определенного, высокого уровня. Да, он должен проверять то, что реализуют программисты, на соответствие требованиям к автоматизированной

системе и ее частям, но эти проверки – к общей пользе. Так же как и контроль, которому подвергают деятельность его самого в независимых экспертизах качества.

Решение проблемы по устранению искажений в первую очередь связано с организацией коллектива «строителей АС», сбалансированного по функциям участников. А во вторую очередь в построении системы профессиональных отношений, при которых неизбежные искажения отдельных участников проекта органично, хотя и не без труда, будут исправляться. Затем на эти отношения можно будет «навешивать» ту или иную методику проектирования АС, выбирать инструменты моделирования или отказываться от них.

Исходные требования к автоматизированной системе всегда противоречивы, часть из них будет отвергаться кем-то из участников проекта. Но никому из них не стоит «тянуть одеяло на себя», ведь пройдет время – несколько месяцев или лет – и спорное требование будет реализовано, пусть другими людьми, частично или в иной форме.

3.4.6. Типовые подходы и типичные проблемы

Методология построения АС содержит три основные составляющие.

- *Набор моделей* (если строго, то типов моделей) для описания требований к АС, проектных и программных решений. Каждая модель обычно содержит средства для определения конструкций (потацию) и правила их использования (синтаксис).
- *Методика применения набора моделей для построения АС* обычно использует фиксированный набор моделей и определяет последовательность их построения для описания различных аспектов создаваемой автоматизированной системы.
- *Процесс организации проектных работ* включает различные технологии – планирования, управления проектом, контроля качества и т.д.

Каждый проект можно рассматривать как реализацию конкретного процесса применения методики. В зависимости от ограничений по срокам и стоимости в конкретную реализацию могут быть включены лишь отдельные части полной методики и процесса.

Методики описания АС обычно относят к одному из двух типов: структурному или объектному. Правильнее было бы говорить о структурном и объектном наборах моделей, так как существуют методики построения одних и тех же моделей с несовместимыми синтаксическими правилами.

Структурная методика обычно предполагает наличие пяти компонентов.

- *Диаграмма потоков данных/модель технических – процессов (Data Flow Diagram/Business Process Model)* была предложена в конце 1970-х годов как средство описания процессов обработки информации. Основные элементы модели: процесс, поток и хранилище, представляющие соответственно обработку, передачу и хранение данных (или материальных объектов). Контекст работы автоматизированной системы представлен с помощью внешних сущностей. В настоящее время эту модель используют в основном для описания технических процессов (производственной деятельности).
- *Диаграмма «сущность – связь» (Entity Relationship Diagram)* была предложена в середине 1970-х годов как средство описания информационной модели предметной области, не привязанное к инструментам реализации структур хранения данных в автоматизированной системе. Элементы модели: сущность и связь, представляющие типы объектов предметной области и их отношения.
- *Диаграмма переходов состояний (State Transition Diagram)* в основном используется для моделирования автоматизированных систем реального времени. Основными элементами модели являются состояние (объекта или системы) и переход из одного состояния в другое. Можно использовать эту диаграмму для документирования состояний как программных конструкций (экранов, каналов связи), так и объектов реального мира (выполняемых заказов, производимой продукции).
- *Структурная карта (Structure Chart)* показывает, как программные модули (функции) вызывают друг друга в процессе выполнения программы. Основные элементы – программный модуль и вызов. Для вызова могут быть указаны передаваемые и возвращаемые параметры. Также существуют обозначения вызова в цикле, условного вызова и лексического включения модулей. Структурная карта служит основной моделью для описания структуры программного кода на универсальных языках программирования третьего поколения.
- *Структурные схемы (Flow Chart)* содержит вычислительные блоки, используемые дисциплиной структурного программирования. Описывает алгоритмы выполнения процедур.

Структурную методику характеризует раздельное построение модели функций (чаще всего диаграммы потоков данных) и модели данных (чаще всего диаграммы «сущность – связь»). К сожалению, авторам различных «школ» так и не удалось договориться об единой нотации и правилах построения этих моделей. Поэтому большинство CASE-систем, обеспечивающих построение моделей согласно

структурной методике, являются закрытыми и несовместимыми с другими аналогичными системами. Не удалось также сформировать общепринятый стандарт передачи информации между CASE-репозиториями. Невозможность переноса моделей между инструментами разных производителей служит препятствием к внедрению технологий повторного использования изделий, созданию библиотек стандартных решений для разных видов деятельности, интеграции средств моделирования со средствами планирования, управления проектами, тестированием и т.д.

Объектная методика базируется на следующих шести компонентах:

- *Диаграмма вариантов использования (Use Case Diagram)* перечисляет требования, которые должна обеспечить система (информационная или производственная). Основные элементы модели – роли (пользователи системы) и вариант использования автоматизированной системы (предоставляемый сервис). Если речь идет о программном обеспечении, то варианты использования с некоторым приближением можно считать автоматизируемыми операциями.
- *Диаграмма взаимодействия (Interaction Diagram)* описывает порядок взаимодействия участников (объектов) в процессе реализации варианта использования системы. Существует две разновидности диаграмм взаимодействия – диаграмма последовательности (Sequence Diagram) и диаграмма сотрудничества (Collaboration Diagram). Эти диаграммы фактически представляют одну и ту же информацию в разном виде и часто могут быть автоматически получены друг из друга в объектных CASE-системах. Диаграмма взаимодействия вместе с диаграммой вариантов использования применяется при описании как программных, так и производственных систем.
- *Диаграмма классов (Class Diagram)* является основной моделью объектной методике. В некотором смысле ее можно считать развитием диаграммы «сущность – связь». Классы также представляют сущности описываемой автоматизированной системы (программной или производственной), но обладают дополнительными свойствами – поведением, абстрактностью, стереотипом. Связи между классами также значительно богаче, имеется наследование свойств, агрегация, реализация и многое другое. Основное применение модели классов – описание программного обеспечения, хотя эта модель подходит и для описания производственных систем.
- *Диаграмма состояний (Statechart Diagram)* немного модифицированная диаграмма переходов состояний. Служит для

спецификации состояний объекта, меняющихся в зависимости от развития событий.

- *Диаграмма действий (Activity Diagram)* представляет собой модель, похожую на структурную схему программы. Показывает последовательность выполнения операций (действий).
- *Диаграмма реализации (Implementation Diagram)* показывает структуру созданного программного обеспечения. Имеется два вида диаграмм реализации: диаграмма компонентов (Component Diagram) показывает структуру исходного кода, диаграмма размещения (Deployment Diagram) – структуру исполняемого программного обеспечения.

Для объектной методики характерен набор моделей, связанных с понятием класса/объекта, объединяющего данные (состояние) и поведение. В настоящее время наиболее естественно применять набор моделей, входящих в UML (универсальный язык моделирования). Этот язык стандартизован, широко используется и постоянно развивается. Распространенность языка UML можно объяснить тем, что он создан авторами трех самых известных в мире объектных методов (OMT, OOSE и Booch method). Прекращение «войны методов» и объединение ведущих специалистов привело к открытости и стандартной интерпретации моделей. Стандарт UML открыт для обсуждения и развивается при участии ведущих технологических фирм: Microsoft, Hewlett-Packard, Oracle, IBM, Platinum Technology, Rational Software и др. Использование единого языка моделирования позволяет специалистам разных стран «говорить» на одном языке. На таком языке удобно составлять технические спецификации на коммерческие программные продукты. А стандартизация моделей облегчит интеграцию средств моделирования с другими инструментами.

3.4.7. Сравнение основных моделей – плюсы и минусы

Для описания технических процессов организации достойной альтернативы диаграмме потоков данных пока не найдено. Конечно, эта модель обладает рядом недостатков, самым главным из которых является невозможность показать последовательность выполнения функций, если они входят в несколько технических процессов. По определению все процессы диаграммы потоков данных выполняются асинхронно, и каждый процесс иерархически вложен ровно в один родительский процесс. Для спецификации последовательности выполнения процессов нужно использовать особую разновидность модели – диаграмму функциональной зависимости.

Однако диаграмма потоков данных обеспечивает наглядность преобразования входных данных в выходные, полную картину выполнения совокупности взаимодействующих процессов и

возможность детально специфицировать выполнение каждой операции. И поскольку в настоящее время большое значение приобрели технологии анализа и оптимизации технических процессов, а перед созданием большой автоматизированной системы необходимо понять и описать ее место и роль в общем техническом процессе организации, диаграмма потоков данных своей актуальности не потеряла.

Диаграмма «сущность – связь» подняла описание информации на качественно новый уровень. Аналитики получили возможность говорить о данных в терминах сущностей предметной области, а не структур хранения автоматизированных систем. В настоящее время имеется более богатая и выразительная модель описания данных – диаграмма классов. Однако диаграмму «сущность – связь» нельзя списывать со счетов. У нее есть одно большое достоинство: конструкции этой модели понятным образом соответствуют конструкциям реляционной модели данных. Наличие единого формализма для описания как сущностей предметной области, так и компонентов реляционной базы данных делает диаграмму «сущность – связь» удобным средством для описания хранимых в информационной системе данных. А простота преобразования модели в структуру базы данных (генерация SQL-кода) и обратно (реверс-инжиниринг базы данных) делает данную модель полезной в процессе сопровождения и развития базы данных на сервере реляционной СУБД.

Диаграмма вариантов использования (use case) очень удобна для описания функциональности автоматизированной системы. Каждый вариант использования – определенный сервис, который должна обеспечить автоматизированная система. В этих понятиях удобно описывать, что должна делать автоматизированная система, что нужно тестировать, что принимать у исполнителя. Также обеспечивается трассировка реализации исходных требований к автоматизированной системе.

Большим недостатком данной модели является отсутствие изображения входных и выходных данных, а также информационных связей между вариантами использования. Нет единой картины взаимодействующих функций. Поэтому, как средство описания технического процесса, модель вариантов использования имеет существенные ограничения. Это скорее структурированный список операций, чем модель взаимодействующих функций. Данная модель не может без потери полноты описания заменить диаграмму потоков данных. Каждая из этих моделей обладает своими преимуществами и недостатками. Однозначно выбрать одну из них невозможно. Выбор зависит от того, что в первую очередь пользователь собирается описывать – комплекс взаимодействующих процессов или перечень реализуемых сервисов.

Среди рассмотренных моделей диаграмма классов имеет наибольшее число выразительных возможностей. Это помогает при описании автоматизированных систем, но в то же время делает модель сложнее для построения и понимания. Диаграмма классов удобна при описании как предметной области, так и информационной системы. Следует однако заметить, что модель классов программного обеспечения не является развитием модели классов предметной области. Эти модели состоят из принципиально различных классов, существующих в разных мирах – реальной жизни и компьютерной программе.

Следует четко различать модели как средство описания и методы их использования. Не следует недостатки отдельных методов распространять на используемые в них модели. Модели на основе структурной методике еще далеки от устаревания и определенная ниша их использования будет сохраняться и в будущем. Объектные модели позволили описать многие аспекты, перед которыми структурные модели бессильны. Но нельзя считать объектные модели современной заменой моделей структурных. Они также обладают рядом недостатков, ограничивающих их применимость. Пока рано говорить о наличии действительно универсального языка описания предметных областей и автоматизирующего их программного обеспечения. Нужно использовать все доступные средства. Чем богаче выбор таких средств, тем лучшего результата можно добиться.

3.4.8. Применение моделей

Каждая из моделей имеет уникальные свойства, отсутствующие у других. И если качество работы улучшится при включении в проектные спецификации какой-либо модели, то нет причин от нее отказываться. Совершенно непонятно, почему нужно делить эти полезные средства на два множества (структурные и объектные) и объявлять их не только несовместимыми, но противоречащими друг другу. Ведь, отказываясь от удобных сочетаний моделей разных методик, мы делаем системные спецификации менее выразительными.

Наиболее правильным является рассмотрение всей совокупности моделей как единого набора инструментов. При построении каждой АС нужно выбирать те инструменты, которые больше всего подходят в данном конкретном случае, не сковывая себя рамками одного метода, каким бы популярным он ни был. Например, можно выбрать модель технических процессов для описания автоматизируемой деятельности, диаграмму вариантов использования и диаграммы взаимодействия для описания состава ПО, диаграмму «сущность – связь» для описания реляционной базы данных, структурную схему для описания наиболее сложных фрагментов кода.

В качестве одного из главных аргументов против сочетания разных моделей выдвигается необходимость смены формализма при переходе от одной модели к другой. А так как обычно нет однозначного соответствия между конструкциями разных моделей, то нет и типового алгоритма получения одной модели из другой. Следовательно, этот переход требует работы квалифицированного специалиста. Приходится раз и навсегда отказаться от заманчивого мифа пайти инструкции, по которым сложные автоматизированные системы могли бы быть созданы без участия высококвалифицированного персонала. Это невозможно. Проектные решения рождаются в головах специалистов. Моделирование – это лишь способ упорядочить и зафиксировать принятые решения.

Заманчиво также было бы иметь единый способ описания различных аспектов АС и процесс ее создания свести к последовательному развитию одной модели. Некоторые сторонники объектных технологий такой моделью объявили диаграмму классов. Стремление все описать классами вновь создает ситуацию молотка в руках. Само наличие такого единого решения вызывает сомнения: слишком сложен и многогранен предмет описания.

Следует всегда помнить, что единственное место, где рождается автоматизированная система, это головы ее разработчиков. Какие бы методы разработчики ни использовали, от необходимости принимать проектные решения они вас все равно не избавят.

Что же делать руководству предприятий в текущей ситуации? Наши соображения по этому поводу можно изложить в виде следующих тезисов.

1. Сохранение и повышение на своем предприятии культуры работы вообще и культуры управления в частности. Для этого следует поощрять усилия по созданию и описанию деятельности в виде технических процессов при решении часто используемых проектных задач. А это в свою очередь станет «каркасом» для правильных и легко прослеживаемых действий отдельных исполнителей. Замечено, что компьютеризация «зомбирует» исполнителей на правильные действия, т.е. использование АС позволяет снизить уровень беспорядка на предприятии.

В противном случае предприятие будет разоряться, поскольку более дальновидные конкуренты вытеснят его с рынка из-за больших издержек, дорогих товаров и низкой культуры обслуживания, т.е. предприятие больше потеряет при работе с заказчиками из-за ошибок собственных исполнителей.

2. *Информационные технологии следует рассматривать и использовать* не только как средство для подъема культуры и квалификации персонала, но и *как важный инструмент сохранения*

«боеготовности» предприятия для создания лучшей продукции, выпуска лучших товаров. В противном случае вы больше потеряете во время попыток освоения новых основных технологий. Как показывает практика последних лет, лучшая продукция часто и неизбежно будет связана с новыми информационными технологиями.

3. Возможности в управлении предприятием во многом определяются применением информационных технологий и качеством АС предприятия. Как показывает практика, качество – это один из ответов на любой кризис. Создание по настоящему качественной АС под свои пужды и для решения своих задач может стать тем катализатором, с помощью которого может быть реанимировано существующее производство.

3.4.9. Пояснение применяемой терминологии

CASE-система – программный комплекс для описания предприятия, информационной системы и/или генерации различных частей автоматизированной системы.

CASE-репозиторий – база данных **CASE-системы**, в которой хранится проектная информация, представленная сложными взаимосвязями данными – графическими диаграммами; программными спецификациями и др.

Каскадный (водопадный) жизненный цикл – поэтапное, последовательное построение автоматизированной системы. Каждая стадия (обычно это анализ, проектирование, программирование, тестирование, внедрение) полностью завершается перед началом следующей.

Спиральный жизненный цикл – многократное прохождение стадий построения автоматизированной системы. Возможность возврата на начальные стадии позволяет учитывать изменяющиеся требования к системе.

Функциональная декомпозиция – разбиение описания деятельности на основе выполняемых функций. Выделяются виды деятельности, затем основные процессы, затем детализируется их выполнение.

Диаграмма функциональной зависимости – графическое изображение последовательности выполняемых функций. Описывает процессы и события, генерируемые этими процессами и иницирующие их.

Реляционная модель данных – представление данных в виде наборов атрибутов (реляционных отношений), над которыми можно корректно выполнять некоторые хорошо формализованные операции (выборку, соединение, проекцию и т.д.).

Реверс-инжиниринг – восстановление спецификаций существующей автоматизированной системы.

Метод OMT (Object Modeling Technique), предложенный Джеймсом Рамбо (James Rumbaugh), – метод описания автоматизированной системы, заключающийся в построении диаграммы потоков данных для описания выполняемых функций, а затем создания на ее основе модели классов для программной реализации.

Метод Буча (Booch method) – чисто объектно-ориентированный метод описания автоматизированной системы, предложенной Гради Бучем (Grady Booch).

Метод OOSE (Object – Oriented Software Engineering) – метод описания автоматизированной системы, предложенной Иваром Якобсоном (Ivar Jacobson). В рамках этого метода была предложена диаграмма вариантов использования (Use Case Diagram), вошедшая в стандарт UML.

3.5. Возможности по созданию систем автоматизированной поддержки информационных решений силами пользователей

В главе рассматривается подход к созданию систем автоматизации поддержки информационных решений (САПИР) на примере задач конструкторско-технологической подготовки производства (КТПП). В качестве CASE-средств для создания САПИР использованы отечественные программные инструментальные системы.

3.5.1. Роль и место систем КТПП в обновлении выпускаемой продукции

В последние годы наблюдается колоссальный рост потребности промышленности в системах *конструкторско-технологической подготовки производства*, охватывающей решение задач по всему циклу производства (дизайн, детализировочное конструирование, технологическая подготовка производства, создание программ для станков с ЧПУ, управление параллельно-последовательным проектированием и т.д.). При этом в промышленности прослеживается устойчивая тенденция неуклонного роста срока жизни систем КТПП в рамках создаваемых объектов. Это свидетельствует о том, что функциональные возможности систем КТПП стабилизировались и отвечают текущим потребностям пользователей. Обзор существующих зарубежных графических систем для решения задач КТПП на базе рабочих станций и их возможности приведены в работе.

За время активной эксплуатации автоматизированных систем конструкторско-технологической подготовки производства практически неизбежны серьезные изменения в структуре самой организации, в вопросах экономии средств, сокращении сроков запуска в производство новых объектов, оптимизации производственных

процессов и т. д. Все это необходимо учитывать при выборе и приобретении базовой автоматизированной системы КТПП организации.

Это новое отношение к выбору и приобретению автоматизированных систем КТПП обусловлено следующими доводами:

- *руководство предприятия осознало, что на нынешнем этапе знания представляют собой редкий и дорогой ресурс;*
- *повышены требования к качеству и сложности создаваемых объектов;*
- *ужесточены ограничения на допустимые затраты и время, необходимое для выхода на рынок с новым продуктом;*
- *амбициозность целей производителей неуклонно растет.*

Все эти доводы заставили по новому отнестись к архитектуре построения автоматизированных систем КТПП в целом и по новому переоценить место и роль базовых инструментальных средств автоматизации проектирования, с использованием которых системы КТПП созданы. Практика показала, что только при помощи инструментальных средств, основанных на CASE-технологии, можно обеспечить быструю адаптацию к постоянно меняющимся требованиям в промышленности.

Рост насыщенности производственного процесса средствами вычислительной техники, появление компьютерных сетей, кооперация работ в рамках «виртуального предприятия» также вызвали серьезные изменения во взаимоотношениях между партнерами, клиентами, подрядчиками и субподрядчиками. Часто возникает ситуация, когда участники производственного процесса вынуждены использовать разные программные системы. А это в свою очередь создает проблему обмена данными, необходимыми для решения поставленных задач. Это означает, что средства конструкторско-технологической подготовки производства должны обладать возможностью адаптации к меняющимся условиям, а участники производственного процесса – владеть оптимальным способом управления этими изменениями в конкретных условиях. Все это указывает на то, что системы КТПП должны быть ориентированы не только на моделирование создаваемого технического объекта, но и на связанные с ним процессы поддержания жизнедеятельности производственной среды по его созданию.

3.5.2. Ограничения традиционных систем КТПП

Архитектура используемых ныне систем компьютерной подготовки производства была разработана в середине 1980-х годов и основана на возможностях универсальных операционных систем. С тех пор существенно изменились базовые требования к автоматизированным системам КТПП, что не нашло пока должного отражения в архитектуре программных средств. Эти изменения затрагивают

различные аспекты разработки и функционирования автоматизированных систем КТПП. Остановимся на некоторых из них.

- *Процессы.* Первоначальные требования производства были ориентированы на создание и решение актуальных и наукоемких для своего времени отдельных задач (трехмерное проектирование, анализ, изготовление). Сегодня же акцент ставится на компьютерное сохранение и использование полной информации об объектах на протяжении всего цикла проектирования и изготовления (примером тому служит возрастающий интерес к CALS-технологии).
- *Функции.* В 1980-е годы ядро КТПП основывалось на обработке геометрических моделей – проволочных, поверхностных и твердотельных. Для развития этого подхода применялись специальные средства, такие как параметрическое и адаптивное моделирование. Основу этих средств составляют методы начертательной и аналитической геометрии, в основном охватывающие пространственно-размерные связи. В настоящее время информация, описывающая объект, должна включать в себя данные не только о пространственно-размерных связях, но и о свойствах материала, информационные, временные, экономические связи (например, о марке материала, допусках, размерных связях, результатах прочностного анализа, о траектории инструментов в процессе обработки и т.д.). При этом вся эта информация должна быть тесно переплетена с информацией о геометрии. То есть используемая сегодня информация базируется на электронной модели объекта, возможности которой позволяют получать как комплект традиционных чертежей, так и использовать ее для компьютерного моделирования функционально важных свойств объекта (взамен натуральных испытаний), обработки деталей на оборудовании с ЧПУ и т.д. Таким образом в настоящее время в КТПП наблюдается переход от методов аналитической и начертательной геометрии, в которых никак не учитываются отклонения, шероховатости, свойства материала и т.д., к электронным моделям объекта (развитие этого подхода нашло свое отражение в стандарте STEP).
- *Архитектура построения автоматизированных систем КТПП.* В традиционных системах САПР/АСТПП все «поу-хау» сконцентрировано в программах, а не в данных, с которыми они работали. Для того чтобы решить новую задачу, следовало разработать новое приложение с новой моделью данных. Согласно современным воззрениям на технологию программирования модель объекта должна интегрировать в себя все это «поу-хау». Суть различий в архитектурном

построении «старых» и «новых» автоматизированных систем КТПП можно продемонстрировать с использованием следующей аналогии. Архитектура ЭВМ типа IBM360/370 и их аналогов строилась вокруг центрального процессора, к которому подключались все остальные устройства. Архитектура персональных ЭВМ использует принцип информационной шины. Новая архитектура при построении компьютера позволила стандартизовать основные интерфейсы и организовать массовый выпуск ПЭВМ. Это в свою очередь резко снизило цены на них и обеспечило в короткий срок доминирующее положение на рынке средств вычислительной техники. В программных системах переход к новой архитектуре происходит в настоящее время. Практически во всех поставляемых на рынок системах КТПП центральное место занимает база данных, вокруг которой средствами компьютерного моделирования реализуются функциональные действия автоматизированных систем КТПП. Устройство базы данных и особенности работы с ней у каждого поставщика индивидуальны. Архитектура систем КТПП нового поколения строится на базе «информационной шины». Описание технического объекта унифицировано и стандартизовано.

- *Затраты.* Рабочие станции для КТПП в начале 1980-х годов стоили примерно в десять раз дороже, чем сейчас. Та же тенденция наблюдается в приложениях: рост их мощности при одновременном снижении цены и повышении качества. Практически все фирмы, поставляющие системы КТПП, объявили о своей поддержке платформы Wintel.

Радикальные перемены в производстве повлекли за собой появление таких технологических требований, которые превосходят возможности программной архитектуры и техники традиционных систем: структуры данных должны быть открытыми и способными к эволюции, геометрия и технология изготовления должны сосуществовать в одной и той же архитектуре, качество программного обеспечения – повышаться, а стоимость программ – снижаться. При этом сложность создаваемых технических объектов постоянно возрастает. Все это предполагает технологический скачок в автоматизированных системах КТПП, невозможный без переработки архитектуры традиционных программных средств.

3.5.3. Отличия «старого» и «нового» понимания КТПП

Основные отличия «старого» (в смысле традиционных САПР/АСТПП) и «нового» (САПИР) понимания при решении задач КТПП заключаются в:

- *способах представления знаний в электронной форме о решении задач КТПП в прикладной области;*
- *архитектуре построения системы КТПП;*
- *открытости и функциональной завершенности САПИР;*
- *масштабируемости прикладной системы для конечного пользователя.*

Основные методы решения задач КТПП сложились задолго до появления средств вычислительной техники и имеют свою специфику, в которой можно выделить следующие характерные положения.

1. Для решения задач КТПП необходимо использование знаний и опыта пользователя. В своей работе пользователь оперирует самой разнообразной информацией – множеством признаков для классификации деталей, графическими изображениями, числовыми данными, правилами, отражающими опыт и интуицию экспертов, алгоритмами, неформальными процедурами и т.д.

2. В сложившейся нормативно-справочной среде (НСС) машиностроения (книги, ГОСТы, ОСТы, стандарты предприятий (СтП) и т.д.) функциональные зависимости между свойствами и объектами проектирования чаще всего представлены в виде таблиц, параметризованных графических образов, алгоритмов решений, номограмм и т.д., но не в аналитической форме. Отсюда большинство решений достигается не в результате расчетов, а путем выбора пользователем из функциональных зависимостей, отвечающих заданным требованиям.

3. В ряде случаев решение задач достижимо при неполных, а иногда и противоречивых сведениях, отсутствие формальной организации НСС приводит к низкой эффективности поиска необходимых пользователю фактических сведений. справочная информация, помогающая принимать решения, разнородна и извлекается из разнообразных источников.

4. Часть решений задач реализуется по схеме: предположение – выбор ближайшего подходящего решения – поверочный расчет. Возможные решения многовариантны, критерии для их оценки далеко неоднозначны – пользователь принимает решения в зависимости от своего опыта, знаний и предпочтений.

5. Большой объем конструкторско-технологической документации представлен в алфавитно-цифровой и графической форме, оформление выходной документации регламентировано стандартами (ЕСКД, ЕСТП, ISO и т.д.).

6. Знания и опыт экспертов-проектировщиков, на основе которых формируется большая часть решений, доступны только в форме высказываний на языке деловой прозы. Это требование предполагает наличие

определенной подготовки пользователя по работе со сложившейся НСС и использование специализированного понятийного аппарата.

Наличие такой специфики определяется накопленным за многие десятилетия практическим опытом проектирования из параметризованных графических образов с соответствующими правилами принятия решений. За это время сложилась некоторая форма представления знаний по проектированию объектов машиностроения, которая прошла практическую обкатку специалистами в промышленности.

Квинтэссенция этого опыта нашла свое отражение в отраслевых стандартах и справочниках. В этих материалах в явном виде отсутствуют алгоритмы и нет готовых рецептов решения задач КТПП. В материалах приведены некоторые правила и рекомендации, касающиеся отдельных примеров проектирования, использование которых защищает пользователя от серьезных просчетов. В этих рекомендациях в общем виде регламентирован достигнутый промышленный опыт: на уровне конструкций объектов в целом, на уровне отдельных узлов и деталей. Именно такая иерархическая вложенность позволяет пользователю в полной мере проявить свои творческие способности.

Анализ характерных особенностей решения задач конструкторско-технологической подготовки производства показывает, что создание промышленно тиражируемых автоматизированных систем КТПП с учетом пожеланий пользователя средствами традиционной технологии программирования, ориентированной в основном на работу с алгоритмами и множествами данных в терминах универсальных языков программирования, практически достигло своего предела.

Важнейшим достижением новой информационной технологии является переход от использования универсальных языков программирования типа Си, Паскаля и т.д., доказавших свои возможности при решении математических задач, к языкам спецификаций. При этом форма представления знаний в языках спецификаций максимально приближена к традиционно сложившейся в прикладной области. Поэтому запись и решение задач на языках спецификаций уже доступна не только программисту, но и специалисту прикладной области, способному работать со средствами вычислительной техники.

3.5.4. Системы автоматизированной поддержки информационных решений

При организации производственных процессов было бы желательно сохранить простоту использования, опыт и навыки конструктора по работе со справочником. Переход от хранения знаний в пассивном виде (книга) к активному их использованию в компьютерной среде представляет новые возможности для конструктора. В этом случае сведения из справочника доступны конструктору в виде не

только обычных библиотек с символьными или графическими описаниями, но и интеллектуального помощника, который следит за тем, что и как делает конструктор, дает советы по ходу работы, генерирует параметризованные фрагменты технических решений по исходным данным, при необходимости объясняет принятые решения и т.д. В этом случае конструктор оценивает предлагаемое ПЭВМ решение, при необходимости их редактирует (как символьные значения, так и графические решения) и использует в своей практической деятельности. Принципиально, что вся эта практическая деятельность пользователя, выражаясь научным языком, протекает в условиях нехватки информации и многокритериальной оценки альтернатив.

Подход к организации деятельности конструктора в компьютерной среде реализован в рамках методологии создания систем автоматизированной поддержки информационных решений (САПИР). Суть этой новации, представляющей собой одно из отличий САПИР от традиционных систем автоматизации проектирования (САПР), состоит в том, что каждый конструктор может самостоятельно определить в привычной для него форме «Правила проектирования», сложившиеся в предметной области, и использовать их в повседневной работе. Созданные отдельные «Правила» можно объединять в типовые методики проектирования.

Для удобства пользователей большая часть правил из нормативно-справочной информации по проектированию асинхронных электродвигателей в виде расчетов, баз данных и типовых графических решений предварительно введена в систему. На основе этих сведений разработаны типовые методики для проектирования различных исполнений электродвигателей. Как показывает практика базовый набор правил достаточен для решения большинства задач расчета и конструирования электродвигателей и компьютерной подготовки его производства.

В эти же «Правила» пользователь может самостоятельно включать и ноу-хау своего предприятия. А это в свою очередь обеспечивает накопление знаний и защищает интересы предприятия в случае ухода специалиста. В целом это позволяет повысить качество выполнения работ и избежать серьезных ошибок в процессе проектирования и изготовления.

3.5.5. Особенности построения САПИР

Компьютерная база знаний (КБЗ) САПИР задач КТПП состоит из символьной и графической частей. В состав символьной части КБЗ входит система словарей и справочников прикладной области, таблицы с данными и набор информационных блоков, фиксирующих правила принятия решений. Работа с символьной базой знаний

реализована в рамках компоненты интеллектуальная компьютерная среда ИКС.

В работе специально для решения задач КТПП был предложен и реализован язык спецификаций, построенный на базе блоков принятия решений. На основе материалов из этой работы под руководством авторского коллектива создается символьная база знаний по решению задач КТПП.

В ИКС перечисляются множество объектов и их свойства, используемые в прикладной области при решении задач. Словарь является единым для всей системы. С помощью словаря организуется общение пользователей с системой в стиле «деловой прозы» конструктора.

Отдельные правила принятия решений оформляются в виде информационных блоков, максимально копирующих форму справочника конструктора и легко заполняемую и редактируемую пользователем с помощью стандартных текстовых редакторов. На основе блоков принятия решений (БПР) средствами ИКС автоматически генерируется программа на универсальном языке программирования Си.

Графическая БЗ содержит набор параметризованных графических прототипов чертежей и заготовок к ним. Вместе с ними в состав графической базы знаний входят словарь понятий и таблицы с фактическими значениями параметров для параметризации итоговых чертежей. Взаимодействие символьной и графической баз знаний приведено на рис.3.22., 3.23. Фактические параметры могут быть рассчитаны, выбраны из таблицы с нормативно-справочной средой или установлены в диалоге по требованию пользователя.

Еще раз подчеркнем, что если информация в традиционном справочнике понятна пользователю, то та же информация, записанная средствами универсального языка программирования, становится недоступной для понимания большинства сотрудников КБ. Это делает необходимым для пользователя вовлечение в процесс традиционного создания программных приложений посредника в лице программиста.

В программных системах, разработанных в 1970–80 годы, процесс проектирования обычно зафиксирован жестко и оформлен как неделимое целое. Отдельные функциональные модули, входящие в состав таких программных систем, часто не могут быть использованы вне этого контекста при решении других задач. Принципы объектно-ориентированного подхода, в основу которого положено понятие объекта и связанных с ним свойств и действий, практически не используется в традиционных системах проектирования. В качестве таких объектов при проектировании может выступать расчет крепежа, выбор свойств материалов и т.д. (здесь следует заметить, что в традиционных справочниках такое разделение на объекты достаточно широко используется).



Рис.3.22. Взаимосвязь отдельных компонент ИКС и ИнИС в ходе создания и эксплуатации символьных баз знаний

Выбранный подход к созданию САПИР, связанный с использованием новой информационной технологии и применением новых форм представления знаний для решения задач КТПП, также изменил архитектуру строения последних. Если входные данные и выходные результаты «старых» автоматизированных систем КТПП и «новых» САПИР остались без видимых изменений, то внутреннее построение

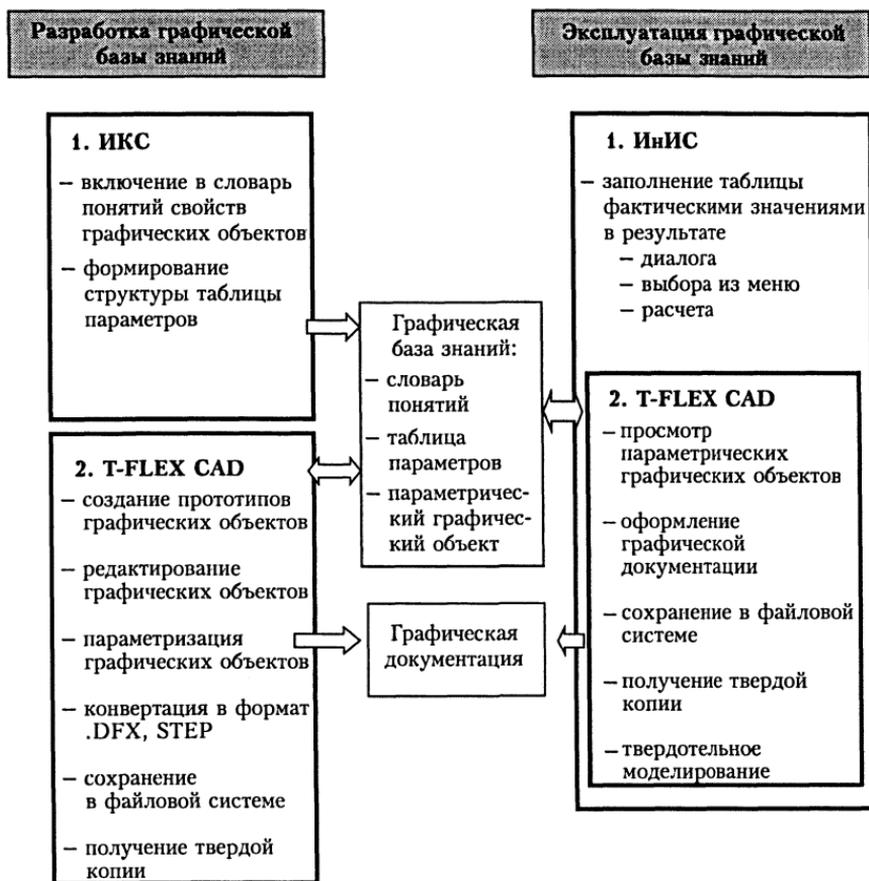


Рис.3.23. Взаимосвязь ИИС, ИКС и T-FLEX CAD в ходе КТПП

системы претерпело серьезное изменение. Это связано с осознанием того, что построить концептуальную модель процесса проектирования одновременно на все случаи жизни достаточно затруднительно. Этап разработки является только началом в жизненном цикле создания автоматизированной системы, а этап модернизации должен быть предусмотрен с самого начала ее создания. Модернизации может подвергаться наполнение НСС, изменение расчетных задач, переход к новой программно-технической среде и т.д.

Новый подход к созданию САПИР повышает гибкость автоматизированных систем. В «новых» разработках вместе с САПИР поставляется некоторая CASE-технология (рис.3.24), которая призвана

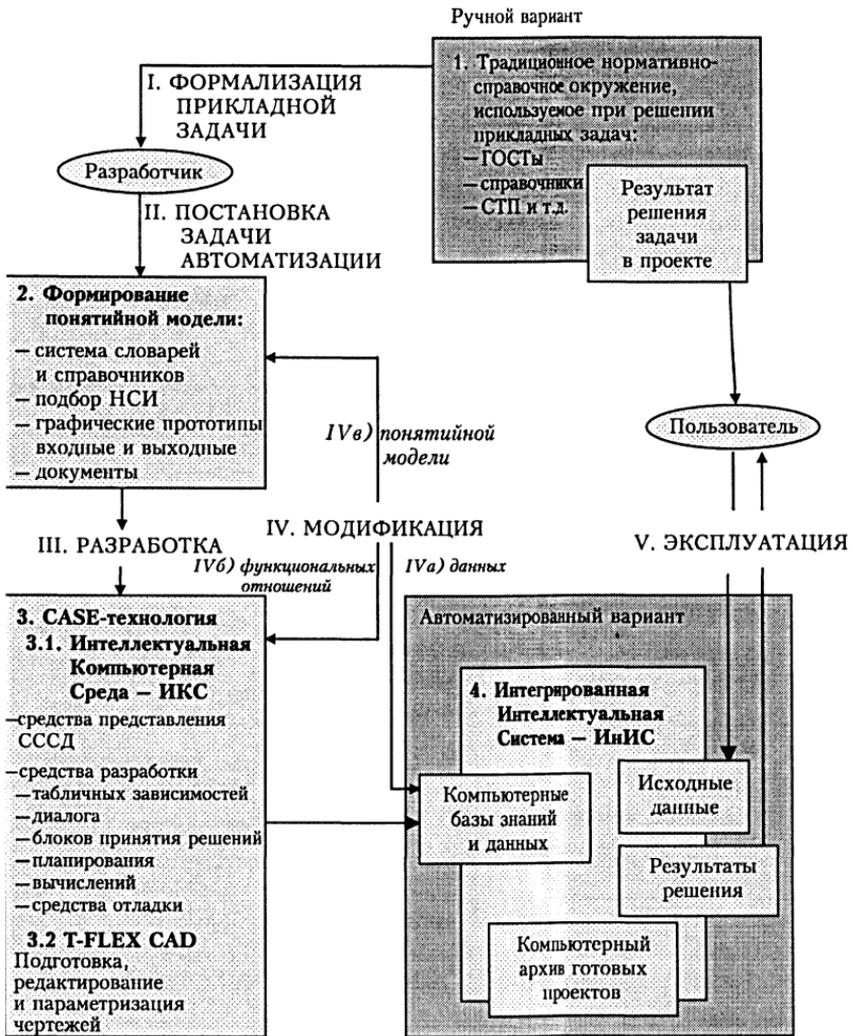


Рис.3.24. Роль и место CASE-технологии в ходе создания и эксплуатации программных приложений пользователя при КТПП

мочь пользователю самостоятельно адаптировать САПИР под свои ли. Эта адаптация может подразумевать различные корректировки автоматизированной системе. Так пополнение таблиц новой нормативно-справочной информацией практически никак не отражается на работоспособности автоматизированной системы и доступно пользователю в любой момент (эта возможность на рис.3.24. помечена как вариант Б.1 при модернизации). Расширение функциональных

зависимостей потребует повторной сборки отдельных вычислительных моделей (вариант – Б.2). В САПИР предусмотрена возможность подключения ранее созданного пользовательского функционала. Внесение пользователем дополнений в понятийную модель потребует изменений в системе словарей и справочников данных и функциональных зависимостей (вариант – Б.3). Статистика показывает, что на отдельных предприятиях соотношение между этими изменениями варьируется в следующих пределах: 70% относятся к пополнению данными, 25% – к расширению или редактированию функциональных возможностей и 5% – к пополнению понятийной модели.

Круг функциональных задач, решаемых конструктором в среде САПИР, приведен на рис.3.25 и включает следующее:

1. Ведение проектов.
2. Управление деятельностью конструктора по реализации типовых решений.
3. Ознакомление с понятийной моделью, включающей словарь понятий, каталог функциональных зависимостей и описание их структуры.
4. Заполнение и редактирование полей в диалоге.
5. Работа с таблицами в диалоге и из программ:
 - найти строку, содержимое которой удовлетворяет указанным критериям отбора (ключам отбора);
 - найти множество строк, содержимое которых удовлетворяет указанным ключам;
 - сортировать содержимое таблицы по набору ключей;
6. Расчеты поверочные и взамен натурных испытаний.
7. Анализ полученных результатов.
8. Работа с 2D и 3D графическими объектами.
9. Построение из нормализованных деталей по типовому сценарию базовых конструкций.
10. Сохранение результатов между сеансами работы и в электронном архиве.
11. Получение твердых копий.

Круг задач, решаемых экспертом в среде САПИР, расширен по сравнению с конструктором и включает:

- a) расширение и модификацию понятийной модели прикладной области;
- b) занесение и редактирование значений в таблицах с нормативно-справочной информацией;
- c) формирование новых прототипов параметрических графических образов для общего применения;
- d) разработка, отладка и подключение новых видов расчета.



Рис.3.25. Функциональные задачи, решаемые в САПИР

Поскольку число имеющихся у конструктора каталогов всегда ограничено, а их содержание может и не подойти в конкретном случае, то будем различать процесс конструирования с использованием САПИР по следующим признакам:

- существует каталог решений для пашей задачи (выбор из архива готовых решений);
- существует каталог решений для другой задачи, сводящейся к тем же эффектам (работа по типовому сценарию операций (по модели, аналогу));
- существует каталог объектов с некоторым множеством решений (работа с библиотекой заготовок и не полностью определенным сценарием операций);
- существуют необходимые каталоги операций (требуется доработка графических объектов);

- существуют необходимые каталоги операций (требуется программирование сценария проектирования).

В рамках САПИР CASE-технология реализуется средствами ИКС (символьная часть) и T-FLEX CAD (графическая часть). В качестве оболочки для создания программных приложений пользователя может служить интегрированная интеллектуальная система ИнИС. Освоение этих средств у пользователя, ранее не работавшего с ПЭВМ, занимает от двух до четырех недель; обычно же бывает достаточно одной недели.

В САПИР принципиальным моментом можно считать то, что эксперт по решению задач КТПП является основным разработчиком моделей объектов. Именно он отбирает типовые решения, формирует словари понятий, специфицирует действия в типовых ситуациях, реализует графические прототипы чертежей к ним для многократного использования.

Новая архитектура построения САПИР позволила выделить некоторую инвариантную программную оболочку, которая специализирована на решении задач КТПП (рис.3.26). Программное обеспечение, реализующее оболочку, может достигать до 60% от общего объема САПИР.

САПИР построена на основе объектно-ориентированной архитектуры, создаваемой с самого начала с четким осознанием неизбежности постоянного развития потребностей и технического прогресса. «Каркас» ИнИС предназначен для поддержки современных приложений и тех, которые появятся в будущем. ИнИС существует автономно от программных приложений пользователя, но без этой системы не сможет работать ни одно из них, поскольку в нем заложены основные концепции системной платформы и реализованы базовые возможности, необходимые для любого программного приложения. По сути ИнИС представляет собой общий знаменатель для отдельных модулей САПИР, которые решают свои специфические задачи, а для выполнения стандартных служб последние обращаются к ИнИС (рис.3.27).

Наличие такой повторно используемой оболочки позволяет резко снизить себестоимость создания прикладной системы КТПП. Так, например, на базе ИнИС построены САПИР электродвигателей, режущего и мерительного инструмента, пресс-форм, холодной объемной штамповки, листовой штамповки и т.д.

Предложенная архитектура – это набор инструментов и правил, определяющих структуру программного обеспечения и представляющих базовые средства для разработки и развития приложений: управления данными, определение объектов и их реализация,

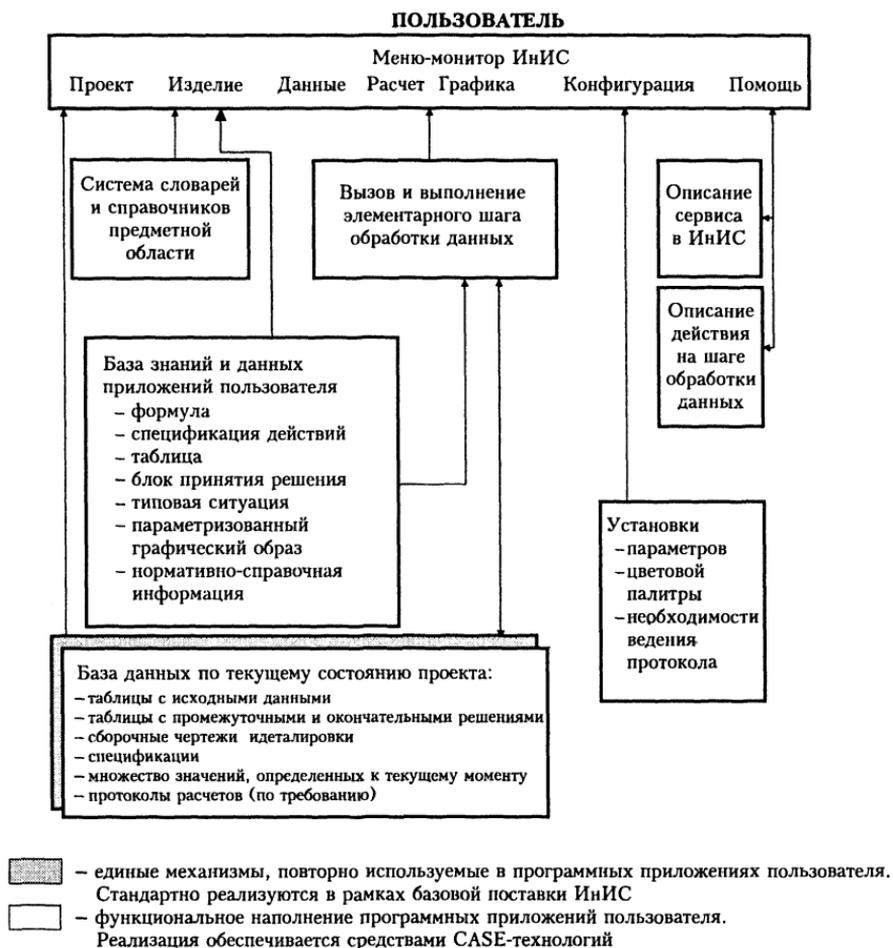


Рис.3.26. Структура типовой «заготовки» системной части программных приложений пользователя в ИнИС.

создание прототипов пользовательского интерфейса, использования визуального программирования, наличие системного сервиса и т.д.

Приложения, созданные на основе ИнИС, обладают рядом преимуществ:

- соответствуют модели «клиент/сервер»;
- поддерживают стандарт распределенных объектов CORBA (OLE);
- демонстрируют явное разделение уровней по определению объектов и их использованию.

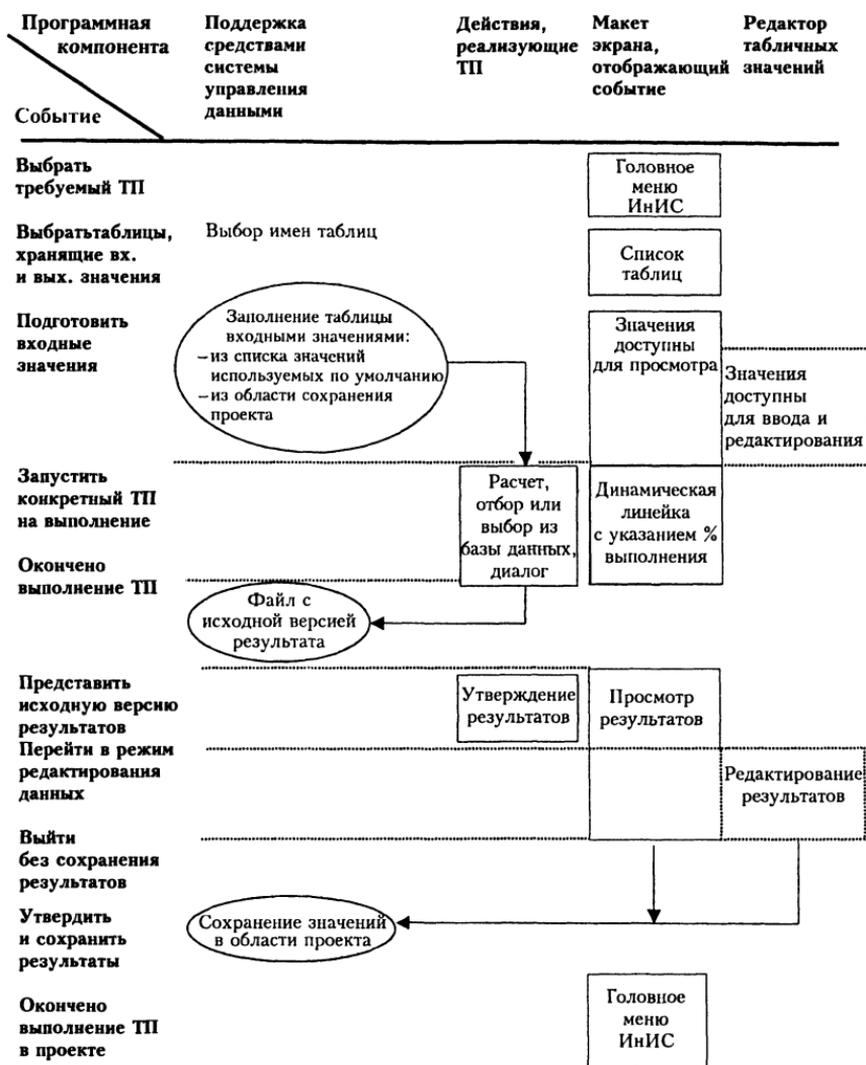


Рис.3.27. Организация программной поддержки событий в ходе выполнения технического процесса (ТП) на ЭВМ

Главное, чего удалось добиться благодаря этой архитектуре, – это открытость и возможность совместного функционирования приложений: поскольку они соответствуют общим правилам, упрощается их интеграция, причем вне зависимости от того, были все они на основе ИНИС или часть из них соответствует OLE.

ИнИС – это также среда разработки, использующая все достоинства объектно-ориентированной технологии. ИнИС демонстрирует выгоды промышленного подхода к созданию программных систем:

- все объекты специфицируются в форме системы словарей и справочников (ССС), формализующей определение объектов и поддерживающих связи между ними;
- все связи между объектами классифицированы и специфицированы;
- автоматическое планирование.

Цель создания такой среды разработки – обеспечение качества программных приложений пользователя с максимальным использованием потенциала среды Wintel.

Содержательная часть информации, интеллектуальные модели объектов, могут передаваться от одного пользователя к другому и наращиваться самими пользователями с минимальным привлечением профессиональных программистов.

При работе с САПИР деятельность пользователя строится по принципу «*Смотри и выбирай*». Этот принцип предполагает, что пользователь:

- *владеет словарем понятий предметной области на уровне «деловой прозы»;*
- *понимает сущность процессов, возникающих при проектировании объектов;*
- *умеет отслеживать и управлять системой с помощью клавиатуры и «мыши» на уровне пользователя Windows;*
- *имеет практические навыки работы с системой графического редактирования и параметризации T-FLEX CAD.*

Результатом проектирования в САПИР для задач КТПП является комплект конструкторской документации, включающий спецификацию, сборочные и детализовочные чертежи, расчетно-пояснительную записку, программы для оборудования с ЧПУ и т.д.

Предлагаемые изменения в архитектуре построения систем КТПП призваны обеспечить следующие преимущества перед традиционным подходом:

- сохранение сделанных ранее затрат;
- эволюционность в развитии систем;
- возможность автоформализации знаний;
- переносимость прикладных систем и данных между различными платформами;
- масштабируемость систем;
- независимость от конкретных технических и программных средств или конкретного поставщика.

3.6. Конструкторско-технологическая подготовка производства в машиностроении

Современная технологическая среда, связанная с проектированием и изготовлением объектов машиностроения, постоянно насыщается средствами вычислительной техники. Исходя из этого факта, следует строить свою деятельность в области разработки программно-технических решений для задач машиностроения согласно следующего принципа: *сделать конструкторско-технологическую подготовку производства (КТПП) в машиностроении доступной как можно большему числу предприятий и пользователей*. Реализации выдвинутого принципа способствуют следующие обстоятельства:

– широкое распространение и использование на машиностроительных предприятиях персональных компьютеров на базе INTEL и Windows (Wintel) при решении основных задач КТПП. В нашем случае это в первую очередь относится к решению задач, связанных с дизайном, конструированием основного объекта и технологической оснастки для его изготовления, компьютерным моделированием, подготовкой управляющих программ для оборудования с ЧПУ, управлением производством;

- беспримерный рост возможностей базовых программно-технических средств как при создании программных систем, так и при их эксплуатации;
- накопленный опыт по созданию и развитию архитектуры открытых систем применительно к производственным условиям.

Но сами по себе благоприятные обстоятельства не обеспечивают достижение конечного успеха. Только активное сотрудничество при решении задач КТПП со сторонними предприятиями и физическими лицами может привести к успеху. Основой для такого сотрудничества является следующее. *Имеется хорошо отлаженный базовый набор CASE-технологий, который позволяет упростить процесс создания и эксплуатации прикладных программных систем автоматизированной поддержки информационных решения (САПИР) задач КТПП*. Данные CASE-технологии в максимальной степени используют потенциал Wintel и базируется на следующих отечественных инструментальных средствах:

- *системная оболочка – интегрированная интеллектуальная система (ИнИС), реализующая функции виртуальной платформы типа FRAMEWORK при решении задач КТПП в рамках предприятия;*
- *система параметрической графики – T-FLEX CAD, призванная обеспечить создание графической базы знаний по*

объектам машиностроения и решение чертежно-конструкторских задач в условиях твердотельного моделирования (данная разработка является программным продуктом российской фирмы «Топ системы»);

- *интеллектуальная компьютерная среда ИКС, обеспечивающая создание и введение символьной базы знаний по решению задач КТПП с максимальным привлечением знаний и умений экспертов из прикладной области.*

Данные инструментальные средства увязаны между собой в ходе создания и эксплуатации программных приложений пользователя (рис.3.24). На основе этих CASE-технологий силами сторонних предприятий (в первую очередь передовых предприятий в той или иной отрасли и ВУЗов) и отдельных пользователей (обычно экспертов по решению конкретных задач) создаются компьютерные базы знаний и функциональные программные приложения для реализации конкретных задач КТПП. Для этого авторы готовы предоставить собственную методологию создания символьных и графических баз знаний (рис.3.22, 3.23) и приложений по решению задач КТПП, обучать пользователей по работе с системой и продвигать совместную программную продукцию на рынке. Такой подход делает доступной для самых широких слоев рынка покупку предлагаемой программной продукции.

Предложенный подход к разработке программных приложений потребовал сделать доступными для всех потребителей единую методологию создания САПИР на базе CASE-технологий. В конечном итоге это позволяет конечным пользователям при необходимости самостоятельно адаптировать программные приложения, созданные с использованием средств ИпИС, T-FLEX CAD и ИКС, под свои нужды. По нашему мнению, именно такая возможность автоформализации знаний может выступать катализатором для быстрого и повсеместного распространения информационных технологий по созданию и эксплуатации различных САПИР для задач КТПП в машиностроении.

3.6.1. Сам себе программист

Для современных тенденциях развития производства характерно возрастание роли ответственности и интеллектуальных действий конечного пользователя. При традиционной технологии информационных работ ответственность внутри предприятия распределяется между двумя категориями пользователей: конструктором и программистом. В САПИР конструктор осознанно и самостоятельно закладывает в компьютерную модель общие принципы построения, характерные для конкретной работы.

Производственная деятельность конструктора во многом основана на опознании текущей ситуации и выполнении цепочки типовых действий. Чем правильнее и быстрее конструктор опознает текущую ситуацию, тем меньше проб и ошибок он сможет совершить в своей деятельности. Именно это свойство отличает интеллектуальную деятельность: постановка глобальных целей и проверка их выполнения при прохождении локальных целей.

С середины 1980-х годов в мире в качестве основного принципа при создании программных приложений в машиностроении используется подход «Сам себе программист». *Этот подход отражает деятельность, связанную с разработкой программных инструментов и баз знаний, вооружившись которыми пользователи самостоятельно могут использовать базовые наработки по КТПП и при необходимости окончательно доработать последние под свои потребности собственными силами.* Этим обеспечивается сохранность знаний предприятия и возможность накапливать и расширять собственные уникальные решения.

Другой особенностью данного подхода является то, что стоимость базовой наработки обычно на порядок ниже, чем стоимость готового решения. Кроме того, практически не возможно найти двух одинаковых предприятий, т.е. каждое из них имеет свои особенности, и поэтому для каждого потребуются внесение изменений в базовую наработку. Простой настройкой параметров отделаться обычно здесь не удастся.

Более того, эти программные инструменты позволяют пользователям самостоятельно создавать специализированные прикладные системы. В рамках этих программных наработок заложены принципиально новые возможности по решению задач КТПП, обеспечивающие:

- *интегрированные решения по проектированию и производству объектов машиностроения в рамках участка, цеха и предприятия;*
- *совмещение во времени процессов проектирования основного объекта и компьютерной подготовки производства для его выпуска, в первую очередь, при блочно-модульном проектировании;*
- *ассоциативную двунаправленную связь между двухмерным и трехмерным проектированием и моделированием в среде Wintel в ходе всего жизненного цикла;*
- *взаимодействие пользователя и САПИР на языке «деловой прозы» с учетом накопленного опыта и требований отечественных стандартов;*

- *возможность автоформализации знаний и данных силами специалистов предприятия при минимальном обращении к разработчикам и профессиональным программистам.*

На этой базе удается реально приобщать к использованию ПЭВМ в машиностроении инженеров и в первую очередь конструкторов, проектировщиков, технологов и т.д., которые традиционно не относятся к профессионалам в области ВТ.

3.6.2. КТПП как часть корпоративной информационной системы предприятия

Организация подготовки производства в рамках предприятия на основе компьютерно-информационной среды призвана обеспечить достижение следующих ключевых целей:

- повысить конкурентоспособность предприятия в условиях постоянного роста требований потребителей к производимым изделиям;
- улучшить процесс проектирования и подготовки производства путем органичного обмена идеями и информацией (это особенно существенно при решении часто повторяющихся задач);
- повысить темп ведения бизнеса как внутри собственного предприятия, так и при кооперации с партнерами.

Опыт показывает, что автоматизация проектирования на предприятии обычно проходит две стадии. На первой стадии решается вопрос автоматизации проектирования отдельных действий и обмена компьютерными чертежами. Следующий рубеж – создание свособразного пространства-континуума проектирования – от изначальной концепции проекта до его реализации и последующего обслуживания. Построение единого информационного пространства для проектирования и изготовления в машиностроении предполагает:

- создание словаря понятий, описывающего объекты и их свойства;
- организацию проектирования на основе моделей;
- синхронизацию деятельности по проектированию основного объекта и технологической подготовки его производства;
- привлечение специалистов прикладной области к использованию достижений информационной технологии.

3.6.3. Проектирование на основе моделей

Карандаш и ватман долгое время служили проектировщику и конструктору практически единственными инструментами в их повседневной деятельности. Поэтому на первых порах от программ САПР требовали, чтобы они были так же интуитивны как карандаш и ватман. Однако использование карандаша вовсе не интуитивно; это павык, которому обучаются и который развивают. Здесь можно

сослаться на удачный пример фирмы Autodeck. Многие дети собирают те или иные конструкции из блоков конструктора «LEGO», но редко встретишь ребенка, умеющего так же убедительно изобразить эту конструкцию с помощью карандаша. Конструктор должен иметь возможность в рамках проекта работать с отдельными объектами, подобно ребенку, работающему с конструктором «LEGO», нежели создавать конструкцию (пусть и твердотельную) из геометрических примитивов.

Предлагаемый подход к решению задач КТПП связан с необходимостью внести «разумность» в деятельность пользователей на начальных стадиях создания объекта. Это в свою очередь призвано облегчить и ускорить процесс проектирования и обеспечить высокое качество конечного объекта за счет использования возможностей полноценного компьютерного анализа проекта. Для этого программные инструменты проектирования должны быть доступными, простыми в управлении и использовании.

В ходе таких процессов более аккуратно учитывается замысел конструктора. Без учета замысла конструктора вид сверху на сердечник статора и ротора электродвигателя – это всего лишь функционально несвязанные наборы параллельных контуров. Но как только конструктор и ПЭВМ будут понимать, что в плоскости этих сердечников должны присутствовать (и при этом не выходить за его пределы) рабочие пазы, отверстия под крепеж и т.д., и что эти элементы могут изменяться по расположению, количеству, размерам в зависимости от исходных данных и электромагнитного расчета. Кроме того, эти геометрические фигуры сами выступают в качестве базовых исходных данных для последующего проектирования (например, штамповой оснастки). Все это вместе взятое придает принципиально новые возможности предприятию по проектированию различных исполнений объектов на заказ с применением ПЭВМ.

Использование моделей объектов позволяет конструктору работать в терминах «деловой прозы», что дает возможность сосредоточить внимание последнего на реализации собственно разработки, а не на вычерчивании геометрии или борьбе с различными программистскими ухищрениями. Помимо того, использование моделей объектов дает возможность конструктору задавать отношения между отдельными свойствами как внутри объекта, так и между разными объектами в рамках разработки. Принципиально, что отношения между свойствами охватывают не только геометрические характеристики, но и многие другие показатели, которые по замыслу конструктора существенны в данной разработке. Все это вместе взятое позволяет наиболее полно зафиксировать замысел конструктора. Например, сгенерированный по правилам проектирования объект «крепеж»,

помещенный в лист статора и/или ротора, автоматически создает отверстие в заготовке. При этом если удалить объект «крепезж», то отверстие тоже исчезнет.

Организация разработки на основе компьютерных моделей объектов – это нечто большее, чем просто геометрия и топология. В модели под конструкцией объекта понимается совокупность признаков, включающая:

- словарь понятий, представляющих наиболее существенные свойства объекта в предметной области;
- состав и структуру конечного изделия (в дальнейшем перерастает в спецификации на конструкцию);
- функциональные требования к отдельным частям конструкторских решений, согласованных между собой через словарь понятий двунаправленными ассоциативными связями;
- взаимную компоновку частей конструкторских решений в рамках некоторой глобальной системы координат, учитывающую накопленные правила комплексирования из типовых решений;
- возможность получения информативно выразительного представления о принятых решениях как в компьютерной форме (включая возможности 3D анимации объекта), так и в форме чертежно-конструкторских документов;
- возможность выполнения всестороннего компьютерного моделирования, обеспечивающего контроль за соотношениями как между отдельными свойствами объекта, так и всего объекта в целом.

Конструкция отдельной детали представляется совокупностью следующих признаков:

- состав (части);
- форма поверхности (элементарная и сплайновая);
- взаимное расположение отдельных частей в рамках локальной системы координат, учитывающей накопленные правила по взаимосвязи между отдельными частями;
- состояние поверхности;
- размеры поверхности;
- марка материала детали и т.д.

В условиях, когда графическое представление взаимоувязано двунаправленными связями через модели отдельных объектов со всеми аспектами разработки, можно говорить о создании разумной производственной модели, обеспечивающей структуризацию производственного процесса на основе средств вычислительной техники. Для конструкции, созданной по данному принципу, упрощается и последующая деятельность технолога. Это связано с тем, что на

конкретные детали в стандарте предприятия обычно существует унифицированный технологический процесс и управляющая программа для станков с ЧПУ, на которые при необходимости ссылается технолог или заполняет в них только переменную часть.

Технология КТПП в рамках предприятия вносит максимум разумности в деятельность специалиста. Проектирование на основе моделей призвано согласовать взаимосвязанную информационную сеть, подготавливающую технические решения в интересах будущих заказчиков. Проекты на основе моделей – существенно мощнее, чем только использование геометрии и топологии. Они обычно включают словарь понятий, условия сочетания отдельных объектов между собой, спецификации, сборочные чертежи, детализировки, расчетно-пояснительные записки, и даже более абстрактную информацию: последовательность сборки, график обслуживания или время транспортировки и т.д. В этих моделях подробно расписана и согласована деятельность отдельных разработчика в тех или иных ситуациях. При этом эта деятельность в отдельных случаях может допускать возможность использования ПЭВМ в виде обычного электронного кульмана. Когда графическое представление информационно и организационно увязано со всеми другими аспектами проекта, можно говорить о наличии разумной электронной модели в целом.

3.6.4. Объекты в проектировании на основе моделей

Использование объектов призвано обеспечить деятельность проектировщика в условиях применения языка «деловой прозы». Так планштамп задается не просто набором векторов, а в объекте «планштамп» присутствует словарь понятий, определяющий высоту, ширину и толщину, с допустимым рядом значений габаритов, материала, условиями применения и т.д. Это дает возможность каждому конструктору сфокусировать свое внимание на реализации собственно проекта, нежели на вычерчивании с использованием электронного кульмана пусть даже и сложной геометрии. Помимо того, использование объектов дает возможность задавать отношения между отдельными свойствами как внутри объекта, так между разными объектами в рамках проекта как на этапе разработки, так и на этапе эксплуатации. Принципиально, что отношения между свойствами охватывают не только геометрические характеристики, но и любые другие характеристики, которые по замыслу конструктора существенны в данном проекте. Вплоть до того, что можно предусмотреть проверку наличия данного объекта на складе.

Все это вместе взятое позволяет наиболее полно передать замысел конструктора. Например, в САПИР для холодной листовой

штамповки для объекта «контур штампуемой детали» предусмотрены следующие действия:

- подбор из нормативно-справочной базы данных минимально допустимых габаритов плиты;
- автоматическое совмещение центра давления контура штампуемой детали и центра плиты;
- нанесение эквидистантного контура в плите с учетом всех технологических требований.

При этом если изменить объект «контур штампуемой детали», то все действия автоматически претерпят изменения.

3.6.5. Совмещенное проектирование

Принципиальным моментом в предлагаемой технологии является синхронизация деятельности в рамках всего проекта по совмещенной разработке основного изделия и технологической подготовке его производства (рис.3.28). Процесс проектирования и изготовления в машиностроении, как правило, является коллективным и охватывает расчетчиков, проектировщиков, конструкторов, инженеров, производителей и т.д. Все они нуждаются в доступе к различным уровням информации для выполнения своей повседневной работы. Но очень часто эта информация бывает нужна также для обслуживания после завершения проекта. За пределами команды проектировщиков информация также должна быть доступна людям, которые могут оказаться и не специалистами.

В нашем случае в качестве основного объекта выступает производство электродвигателей на Сафоновском электромашиностроительном заводе. При получении нового задания на электродвигатель конструктор просматривает электронный архив. Если в архиве отсутствует аналог, то он создает новый проект, включающий формализованное техническое задание (ФТЗ), расчеты, конструирование, технологию. Технология может быть основана как на унифицированных технологических процессах, так и на единичных. В составе технологии предусматриваются и различные виды оснастки: листовая штамповка, пресс-формы, режущий и мерительный инструмент и т.д. При необходимости предусматривается подготовка программ для оборудования с ЧПУ.

Для каждого технологического вида оснастки процесс разработки идентичен процедуре проектирования основного объекта: поиск в архиве, ФТЗ, расчет, конструкция, технология.

На основе использования согласованной схемы процесса компьютерного проектирования основного объекта и технологической оснастки удастся максимально распараллелить работы на

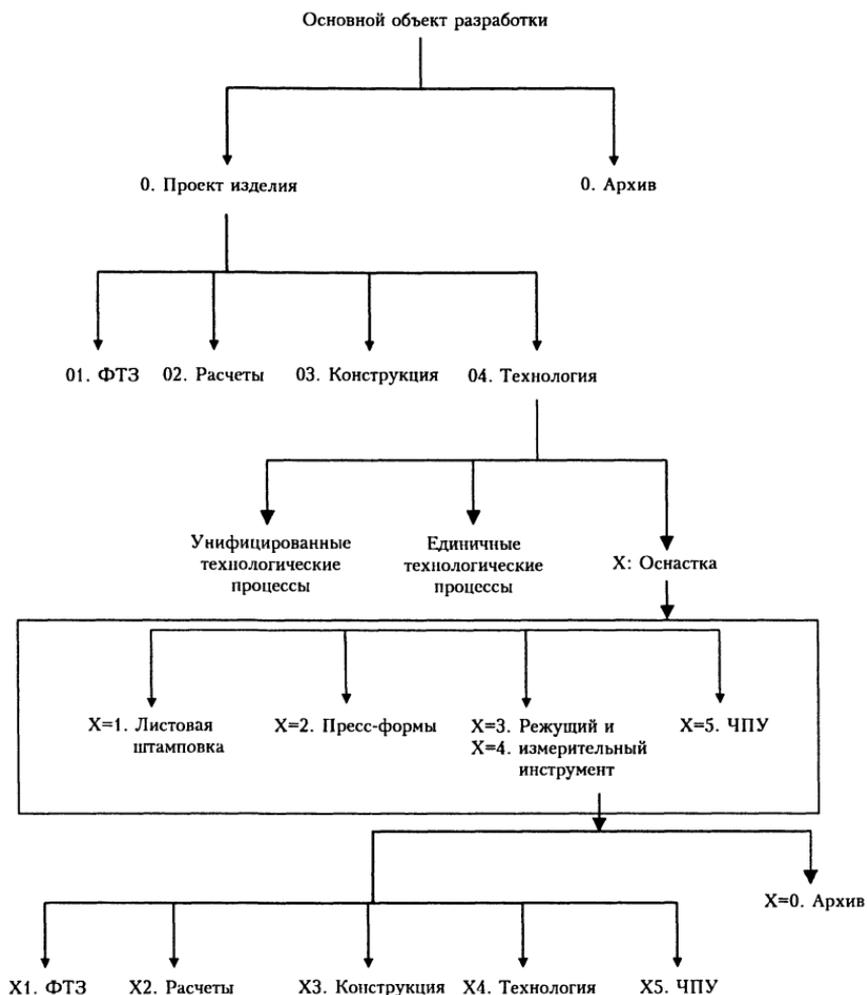


Рис.3.28. Организация совмещенного проектирования при решении задач КТПП

предприятия. А это в итоге обеспечивает сокращение времени выхода нового изделия на рынок.

Изменения на любом из этапов реализации проекта влияют на работу, выполняющуюся параллельно. Такие конфликты обычно разрешаются путем проведения серии согласующих обсуждений, обычно поглощающих много времени. Вследствие таких обсуждений разработки обычно многие отдельные части конструкции должны быть переработаны или, как минимум, расположены по-другому.

Именно поэтому использование чисто графических систем может облегчить модификацию отдельных частей, но они не помогают в решении более глобальных проблем разработки на предприятии в целом.

3.6.6. Блочно-модульное проектирование

В последнее время руководство предприятий все чаще и чаще осознает тот факт, что для проектирования на основе моделей в наибольшей степени подготовлено блочно-модульное проектирование объектов машиностроения. Это связано с тем, что такое производство в основном базируется на известных технических решениях, как правило, состоящих из типовых, нормализованных и стандартных деталей. При этом для блочно-модульного проектирования характерны следующие моменты:

- наличие у объекта одного служебного назначения;
- ограниченное число вариантов технических условий и ограничений;
- использование типизированных правил выбора и расчетов функциональных элементов разных уровней иерархии;
- наличие единой функциональной структуры (или с минимальными изменениями), что в итоге ведет к ограниченности вариантов конструктивного исполнения;
- вариантность исполнения функциональных компоновочных решений, оформленных в виде параметрических рядов.

3.6.7. «Разумные детали» и их коллективное использование

Доступ и многократное использование компьютерных баз знаний по ранее выполненным разработкам образует основной потенциал для возможности роста производительности труда на предприятии при использовании информационных технологий. Подобно тому, как индустриальная революция сделала колоссальный рывок при изобретении стандартизованных и взаимозаменяемых деталей, так информационная революция в проектировании призвана обеспечить рывок с введением в обиход их компьютерных эквивалентов.

Разумная и доступная компьютерная модель по организации деятельности при создании наукоемких изделий может использоваться многократно. А это в свою очередь существенно повышает производительность труда и гарантирует высокое качество.

3.6.8. Простота применения и изучения

Еще одним краеугольным камнем в предлагаемой технологии является простота использования в ходе повседневной деятельности.

В действительности, всегда будут аспекты создания проектной информации, которые не являются интуитивными. Для этих случаев должны быть созданы средства обучения, предназначенные для

минимизации времени, требуемого пользователю для начала продуктивной работы с программой.

Преимуществом такого инструмента является то, что он поддерживает несколько стилей обучения, он доступен круглые сутки и может применяться как для обучения нового пользователя, так и для совершенствования знаний пользователя в какой-либо конкретной области.

Именно поэтому в основу базового интерфейса для наших разработок были положены принципы, используемые в Windows. Поэтому пользователь, умеющий работать в Windows, практически сразу может использовать САПИР и CASE-технологии в своей повседневной деятельности.

3.6.9. Решение задач КТПП в новых условиях

Основная задача КТПП связана с выявлением и разрешением противоречий между требованиями заказчика (функциональными, стоимостными, временными, надежностными и т.д.), вариантами конструктивных решений и конкретными производственными условиями изготовителя. Цена выявления и разрешения указанных противоречий существенно зависит от момента их обнаружения в ходе решения. Поэтому целесообразно стремиться организовать такие условия для выполнения работ, чтобы большая часть этих противоречий могла быть обнаружена и устранена на как можно более ранних стадиях решения, не выходя за стадию информационной проработки конструкторско-технологических решений.

Предлагаемая технология по решению задач КТПП основывается на трех новшествах: это проектирование на основе моделей, совмещенное проектирование основного объекта и технологической подготовки его производства и простота освоения и применения. Три взаимосвязанные технологии, прокладывающие путь новому подходу к решению задач КТПП, включают: *деление создаваемого изделия на типовые объекты, построение единого информационного пространства предприятия и работа в команде над общим проектом*. В этом подходе объекты служат базовыми строительными кирпичиками для проектирования на основе моделей, а модели, функционирующие и развивающиеся в рамках предприятия, облегчают командную работу над проектом.

В этих условиях роль разработчиков сводится к тому, чтобы сделать предлагаемую информационную технологию для решения задач КТПП в машиностроении простой в применении, недорогой и доступной широким кругам пользователей, которые смогут самостоятельно создавать и использовать последнюю в своей повседневной деятельности.

3.7. Опыт использования информационных технологий при проектировании и изготовлении электродвигателей по индивидуальному заказу (на примере ОАО «Сафоновский электромашиностроительный завод»)

Мировой опыт использования информационных технологий при организации производства наукоемкой продукции в машиностроении показывает, что наибольший эффект от их применения могут обеспечить следующие факторы:

- организация и постоянное совершенствование системы управления повторно используемыми компьютерными знаниями в области производства;
- максимальное использование совмещенного проектирования основного изделия и технологической оснастки для его производства, как во времени, так и в пространстве;
- улучшение качества проектирования на основе использования управляемого ассоциативными связями набора типовых моделей базового исполнения, охватывающего весь жизненный цикл изделия;
- информационное взаимодействие и координация всех служб исполнителя под потребности заказчика для выполнения индивидуального проекта в рамках единого информационного пространства;
- переход от натуральных испытаний к компьютерному моделированию как собственно изделия, так и технологической оснастки для его изготовления.

Максимальный эффект от использования информационных технологий на предприятии может быть достигнут только при комплексном применении всех этих факторов и соответствующем управлении мотивацией деятельностью сотрудников.

Основной резерв, на который сегодня могут рассчитывать российские предприятия машиностроительного профиля для повышения своей конкурентоспособности, связан с коренным изменением подхода к организации процесса конструкторско-технологической подготовки производства. Эти изменения в организации деятельности требуют соответствующего методического обеспечения, набора CASE-инструментов, компьютерных баз знаний по основным технологическим переделам машиностроения, учета в ходе выполнения работ международных TQM и CALS-стандартов (рис.3.29).

В работе рассматривается опыт института конструкторско-технологической информатики (ИКИ РАН), МГТУ Станкин и ОАО «Сафоновский электромашиностроительный завод» (ОАО «СЭЗ») по комплексному использованию достижений информационных

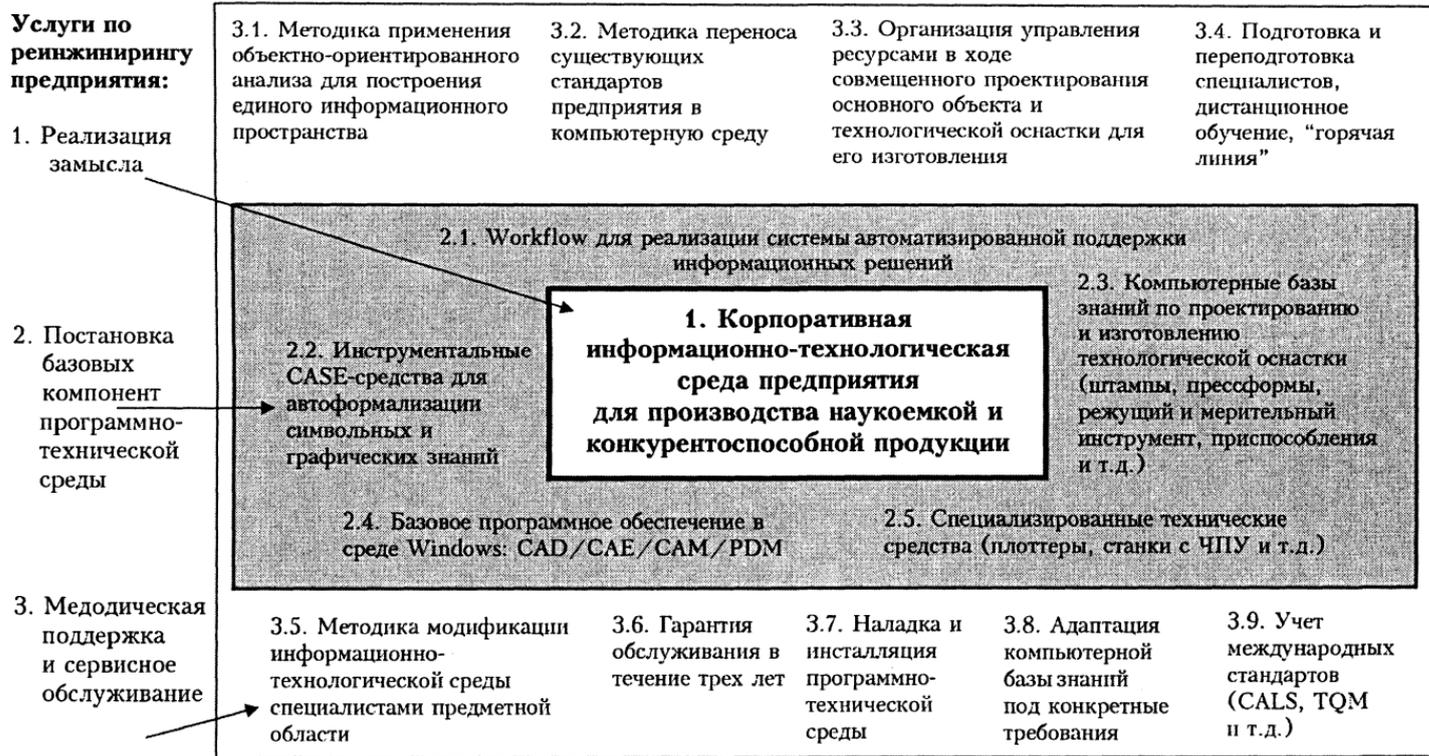


Рис.3.29. Организация создания корпоративной информационно-технологической среды предприятия

технологий при проектировании и изготовлении электродвигателей по индивидуальному заказу в производственных условиях.

Большая часть важнейших технических решений на изделие закладывается на этапе конструкторско-технологической подготовки производства. Плохо спроектированное изделие в производственных условиях может быть только «позолочено», но основные технические решения останутся без существенных изменений. Основная задача конструкторско-технологической подготовки производства связана с выявлением и разрешением противоречий между функциональными, временными, надежностными, качественными, стоимостными требованиями заказчика, допустимыми вариантами конструктивных исполнений и текущими производственными условиями изготовителя. Стоимость выявления и разрешения указанных противоречий существенно зависит от момента их обнаружения в ходе жизненного цикла изделия. Поэтому целесообразно стремиться организовать производственную деятельность так, чтобы большая часть этих противоречий могла бы быть выявлена и устранена на ранних стадиях создания изделия.

На рис.3.30 приведена укрупненная модель деятельности предприятия – исполнителя в ходе реализации индивидуального заказа при мелкосерийной и единичной организации производства наукоемкого изделия (в нашем случае – это электрические машины большой мощности). Заказчик формирует основные исходные требования, которые оформляются как техническое задание на разработку. После заключения контракта для каждого заказа исполнитель формирует проект, в рамках которого выполняется разработка конструкторско-технологической документации на требуемый электродвигатель, реализуется технологическая подготовка его производства и осуществляется собственно выпуск изделия. Выполнение проекта существенно опирается на возможности, предоставляемые компьютерными базами знаний. Наиболее важные взаимосвязи между отдельными составляющими модели приведены на рис.3.30. Собственно реализация заказа в производственных условиях организована с учетом применения MRP и ERP стандартов.

На основании формализованного технического задания с помощью электромагнитных расчетов определяется структура будущей компоновки электродвигателя. Эта структура отражает связь исходных требований заказчика и проектных параметров изделия при конкретной реализации с учетом возможностей производства. В проекте предусматривается пошаговая стратегия разработки решений, которая призвана обеспечить автоматическое обновление компоновочной схемы требуемого исполнения электродвигателя и организовать использование компьютерных средств для детального анализа возможности выполнения в производственных условиях исходных технических требований, сформулированных заказчиком (рис.3.31).



Рис.3.30. Укрупненная модель организации конструкторско-технологической подготовки производства для реализации индивидуального заказа

Пошаговая стратегия разработки решений на основе типовых наборов моделей призвана помочь:

- зафиксировать основные этапы разработки и обозначить множество контролируемых состояний, которые призваны гарантировать получение изделия с заданным качеством;
- описать факторы и приоритеты, влияющие как на график выполнения проекта, так и на функциональные и качественные показатели изделия;
- выявить и учесть критические особенности текущего проекта на ранних стадиях проектирования;
- создать критерии и средства оценки работы групп и отдельных исполнителей;
- организовать взаимодействие как с заказчиком, так и с субподрядчиками на основе взаимоприемлемых компромиссов.



Рис.3.31. Пошаговая стратегия разработки решений при проектировании изделий машиностроения с использованием информационных технологий. *Используемые сокращения:* ЕИП – единое информационное пространство; Γ – главный конструктор; В – ведущие разработчики технических решений (ТР) проекта; И – исполнители проекта

В результате пошаговой стратегии разработки для заказчика повышается «прозрачность» предприятия, улучшается отношение исполнителей к делу, повышается качество принимаемых решений, уменьшается необходимость последующих переделок. Все это в целом положительно сказывается на качестве и цене выпускаемых изделий.

Другим ключевым моментом совершенствования организации деятельности предприятия в ходе перехода к CALS-технологиям становится использование единого информационного пространства, охватывающего весь жизненный цикл изделия. Единое информационное пространство призвано увязать между собой управляемой системой параметров все входящие в сборочную модель электродвигателя компоненты и технологические процессы их изготовления (рис.3.32). Это обеспечивает возможность автоматического обновления компьютерной модели компоновки электродвигателя под требования потребителя. При построении единого информационного пространства учитываются требования STEP-стандарта.

Иерархически процесс проектирования изделия по индивидуальному заказу основан на следующей информационной структуре (рис.3.33). На начальном этапе решение ищется в электронном архиве ранее выполненных разработок. В случае отсутствия в архиве аналога создается проект изделия. В состав проекта изделия включаются следующие папки, в которых создается и сохраняется вся необходимая информация об электродвигателе:

- 01 – формализованное техническое задание (ФТЗ),
- 02 – используемые инженерные *расчеты*, входные и промежуточные результаты,
- 03 – комплект чертёжно-графической документации, содержащей полное описание будущей *конструкции*; в комплект документации входят как твердотельные модели, так и двухмерные чертежи;
- 04 – сведения о *технологии* его изготовления.

Технологическая часть основного изделия проектирования может ссылаться как на единичные, так и на унифицированные технологические процессы. На следующих этапах детализации ($X = 1, 2, \dots$) добавляются сведения о технологической оснастке, требуемой для изготовления данного изделия. В настоящее время поддерживается проектирование технологической оснастки, касающейся штампов для листовой штамповки ($X = 1$), пресс-форм ($X = 2$), режущего ($X = 3$) и мерительного ($X = 4$) инструмента, станочных приспособлений ($X = 5$) и т.д. К технологической оснастке отнесена и подготовка программ для оборудования с ЧПУ ($X = 9$).



Рис.3.32. Задачи, решаемые в компьютерной среде при проектировании и изготовлении электрических машин по индивидуальному заказу

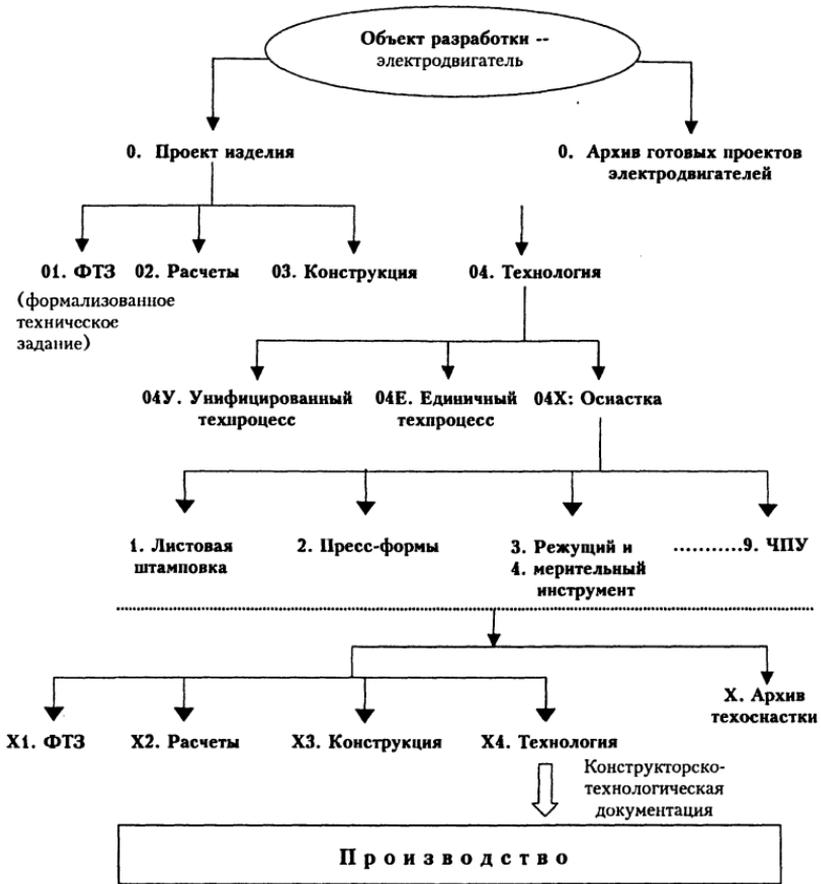


Рис.3.33. Модель организации деятельности при проектировании электродвигателя и технологической оснастки для его производства

Соответственно на уровнях X0, X1, ..., X9 определяется необходимая информация о технологической оснастке, включающая в компьютерной форме архивы ранее выполненных работ. Структурно эти архивы также состоят из ФТЗ, расчетов, конструкций, технологий и программ для оборудования с ЧПУ. Только теперь эти сведения относятся к технологической оснастке. Вся система построена на использовании системы словарей и справочников, образующих единое информационное пространство. Этим достигается возможность отслеживания двунаправленных изменений тех или иных параметров. При этом пользователь будет уведомлен о возможных последствиях предлагаемых изменений.

Изменения параметров на любом из этапов реализации проекта влияют на работы, выполняющиеся параллельно или сделанные ранее. При традиционной организации работ такие конфликты обычно выявляются на рабочих местах в ходе сборки и испытаний и разрешаются путем проведения серии согласующих обсуждений. Вследствие таких обсуждений и согласований отдельные части проекта должны быть переработаны и приведены в соответствие с последними требованиями. Глубина переработки зависит от того, сколь рано выявлена конфликтная ситуация и какие ресурсы уже задействованы при технологической подготовке производства.

При внедрении информационных технологий начальным шагом на пути общего сокращения затрат, повышения качества изделия и предупреждения большинства ошибок является построение единого информационного пространства, набора компьютерных моделей и их согласованное использование на всех стадиях жизненного цикла производства и сопровождения электродвигателя. Существенное сокращение (до трех–четырёх раз) сроков выхода готового изделия на рынок достигается при распараллеливании деятельности по проектированию электродвигателя и подготовке его производства коллективом сотрудников. Для отработки организации и распараллеливания деятельности на ОАО «СЭЗ» использовали концепцию комплексной организации проектирования изделия и подготовки его производства.

Собственно эта концепция не является абсолютно новой для машиностроительных предприятий России. Теоретические основы этой концепции были заложены в работах С.Митрофанова. Только в область действия групповой технологии вовлечена и проектная деятельность предприятия.

Реализация данной концепции на ОАО «СЭЗ» стала доступной с применением системы автоматизированной поддержки информационных решений (САПИР) на принципах CALS-технологий. Эта технология предусматривает использование множества управляемых моделей достаточной степени сложности с произвольным количеством иерархически наследуемых связей. В результате выполнения этих работ стало возможным формализовать в компьютерном варианте научные и производственные знания, накопленные на предприятии в ходе 40 летнего существования, и организовать их повторное использование в дальнейших разработках с применением компьютерных средств.

Большинство производителей инструментальных программных систем для применения в машиностроении декларирует возможность ассоциативности построения взаимосвязанной системы моделей (твердотельная модель, чертеж, расчетная схема, программа для обработки на ЧПУ, технологическая оснастка и т.д.). Но практически у

большинства производителей отсутствует методическое обеспечение, позволяющее реализовать эти возможности на практике, и сведения о том, какую цену должно заплатить конкретное предприятие при внедрении этой концепции.

Опыт внедрения различных графических систем на промышленных предприятиях России показывает, что даже если инструментальная система обладает необходимой функциональностью, то возможность без привлечения сторонних исполнителей справиться собственными силами предприятия с решением данной задачи остается под большим вопросом. Это связано с несколькими причинами.

Во-первых, для организации совмещенного проектирования основного изделия и технологической подготовки его производства число ассоциативных моделей и уровней иерархии может быть очень большим. Построение их собственными силами требует от предприятия много времени и ресурсов. В тоже время базовыми компонентами в ходе этой деятельности выступают процессы проектирования и изготовления технологической оснастки для того или иного машиностроительного передела. Культура создания и производства технологической оснастки на российских и зарубежных предприятиях имеет существенные различия. Простое копирование зарубежного опыта в этой области не всегда может быть эффективным и целесообразным. Это связано с использованием различного оборудования, материалов, стандартных комплектующих, из которых формируется технологическая оснастка, навыками работы и т.д.

Подобно тому как стандартизация и взаимозаменяемость деталей в машиностроении выступили движущей силой индустриальной революции, так информационная революция в промышленности во многом опирается на возможность применения повторно используемых компьютерных баз знаний. Доступ и многократное использование компьютерных баз знаний по ранее выполненным разработкам составляет основной потенциал предприятия для возможности роста производительности труда и обеспечения гарантированного качества. Тиражируемыми компьютерными базами знаний при использовании САЛS-технологий в машиностроении выступают процессы проектирования и производства технологической оснастки.

Во-вторых, совмещенное проектирование должно учитывать индивидуальные процессы производства. Согласно концепции комплексной организации проектирования и изготовления электродвигателя требования формируются в результате выполнения электромагнитных расчетов и должны поддерживать двунаправленные ассоциативные связи с моделью штамповой оснастки, соответствующей технологической документацией, программами для станков с ЧПУ и техническими требованиями заказчика.

Организация совмещенного проектирования электродвигателя и технологической оснастки для его производства призвано согласовать взаимосвязь всех технических процессов еще до начала их использования в ходе практической деятельности вне зависимости от конкретных подразделений и исполнителей. То есть большинство штатных ситуаций, возникающих в ходе выполнения конкретного проекта, не требует от исполнителей консультаций и согласований между отдельными службами и с руководством предприятия. Большая часть решений обговаривается заранее и все ресурсы исполнителя нацелены на практическую работу над конкретным проектом. Эта организация планирования деятельности предприятия построена в соответствии с требованиями TQM-технологии.

Преимущества от применения информационных технологий при использовании последовательной организации работ (левая часть рис.3.34) в общем случае носят локальный характер. В результате достигается небольшое сокращение цикла изготовления при автоматизации существующих задач, некоторое улучшение качества за счет однозначного описания изделия, незначительное преимущество от использования твердотельной модели. Все это в целом оказывает минимальное влияние на цели, стоящие перед предприятием. Именно поэтому использование чисто графических систем, не погруженных в CALS среду, может только облегчить модификацию отдельных частей изделия, но они не в состоянии обеспечить предприятию существенного выигрыша в решении глобальных проблем разработки, подготовки производства и сопровождения изделия в целом.

При совмещенной организации работ по основному изделию и технологической оснастки для его производства происходит полная реорганизация всего жизненного цикла выпуска изделия (правая часть рис.3.34). За счет совмещения во времени процессов проектирования основного изделия и технологической оснастки для его производства удастся существенно сократить сроки выхода изделия на рынок. Такой подход позволяет обеспечить организацию бизнеса в быстром темпе и гарантировать значительное повышение качества выпускаемого изделия. Этим достигаются основные цели применения CALS-технологии в машиностроении.

В нашем случае все управление проектом сконцентрировано в едином информационном пространстве, задающем функциональные характеристики электродвигателя и связывающем отдельные технические процессы между собой посредством многоуровневых управляемых ассоциативных связей. Это позволяет быстро создавать новые изделия по индивидуальным заказам на основе базовой компоновки электродвигателя, а также строить электронное описание изделия в терминах «деловой прозы». Ассоциативная связь, реализуемая

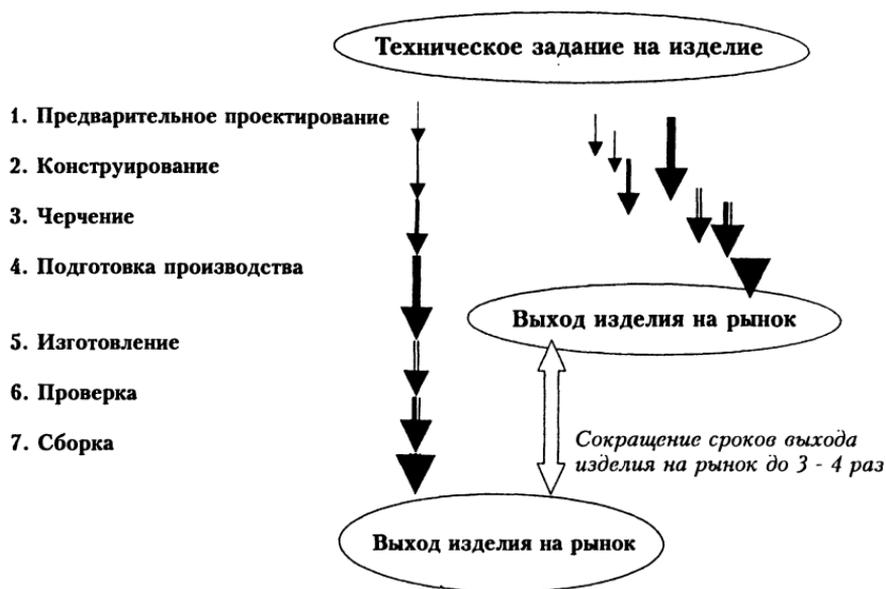


Рис.3.34. Реорганизация процесса выпуска нового изделия

компьютерными средствами в едином информационном пространстве, дает возможность автоматически учитывать изменения по проекту во всех предусмотренных представлениях электродвигателя и технологической оснастки для его изготовления (рис.3.35).

При такой организации разработки более аккуратно учитывается и фиксируется замысел исполнителя в ходе конкретного проекта. Например, без учета замысла конструктора любой чертеж – это всего лишь функционально несвязанные наборы геометрических контуров. Но конструктор и среда с единым информационным пространством должны понимать, что все эти геометрические контуры сами выступают в качестве базовых исходных данных для последующего проектирования (рабочие пазы, отверстия под крепеж и т.д.) и могут изменяться по расположению, количеству, размерам в зависимости от исходных данных и электромагнитного расчета и при этом не выходить за габариты изделия. Кроме того, эти геометрические контуры выступают как исходные данные для последующего проектирования технологической оснастки (например, штамповой).

Все это вместе взятое дает принципиально новые возможности предприятию при разработке различных исполнений наукоемких объектов по индивидуальному заказу в компьютерной среде с применением CALS-технологий. Такой подход предполагает экономически



Рис.3.35. Возможности пользователя при проектировании электрической машины на основе единого информационного пространства.

Используемые сокращения: Т – выбор из таблицы; D – ввод значения в диалог; E – инженерные расчеты; R – графическое редактирование

эффективный и рациональный способ создания, сопровождения и оценки изделий, имеющих общее концептуальное решение и различающихся конкретным исполнением.

Задействовав технологию САПИР при проектировании и изготовлении электродвигателя, разработчики, в случае необходимости, после внесения изменений заказчиком в исходные требования могут оценить что и как будет модифицировано в создаваемом проекте в течение незначительного времени. Это открывает возможность совместной работы заказчика и исполнителя в ходе согласования технического задания, позволяет значительно сократить процесс проектирования и технологической подготовки производства электродвигателей и делает процесс производства хорошо прогнозируемым.

К преимуществам реализованного подхода можно отнести:

- сокращение времени подготовки производства нового электродвигателя с 8000 до 1200 нормочасов за счет организации распараллеливания выполнения технических процессов;
- более тщательное предварительное компьютерное проектирование и моделирование, обеспечивающие нахождение рационального решения без увеличения общего срока создания изделия;
- улучшение качества и сокращение затрат за счет согласования отдельных технических процессов между собой по входу и выходу, используемым форматам данных и т.д.;
- создание и накапливание повторно-используемой компьютерной базы знаний с возможностью ее последующего совершенствования;
- повышение персональной ответственности исполнителей и служб за выполняемые работы.

Таким образом, ряд новшеств характеризует проектирование и изготовление наукоемких изделий на базе САПИР. К этим новшествам следует отнести использование набора компьютерных моделей с типовыми решениями, совмещенное проектирование электродвигателя и технологической оснастки для его производства, простоту освоения и применения пользователем системы в производственных условиях. САПИР предлагает новый подход к решению задач конструкторско-технологической подготовки производства в компьютерной среде. В этом подходе типовые технические решения, накопленные в виде стандартов предприятия, выступают базовыми строительными кирпичиками для организации деятельности исполнителей на основе компьютерных моделей в едином информационном пространстве. Модели постоянно совершенствуются в рамках системы управления повторно используемыми компьютерными знаниями. Все это в целом координирует и облегчает командную работу над проектом.

Информационные технологии, встроенные в систему автоматизированной поддержки информационных решений, позволяют соединить проверенную инженерную практику с широким набором средств компьютерного моделирования. Компьютерная среда, использующая знания об изделии и имеющийся инженерный опыт, значительно упростила решение большинства специальных инженерных задач и обеспечила реализацию прогнозируемого проектирования. А это в свою очередь позволило соединить в создаваемом изделии совокупные знания отдельных исполнителей и подготовить их к последующему использованию в производственных условиях.

Литература

- Аристов Б., Евдокимов С., Киреев В., Кураксин С., Мелешина Г., Рыбаков А.* Деятельность конструктора по блочно-модульному проектированию в компьютерной среде // САПР и графика. 1999. №5. С.73–76.
- Давыдкин А.С., Рыбаков А.В., Ульянов А.М.* Система автоматизированной поддержки инженерных решений при проектировании литейных пресс-форм // Кузнечно-штамповочное производство. 1998. №7. С.42–48.
- Давыдкин А.С., Ульянов А.М.* Система автоматизированного проектирования пресс-форм для литья термопластов под давлением // Автоматизация проектирования. 1998. №3. С.29–32.
- Дмитров В.И.* Опыт внедрения CALS за рубежом // Там же. 1997. №1. С.3–9.
- Евдокимов С.А., Рыбаков А.В.* Программно-компьютерная среда для автоматизации знаний // Вестник машиностроения. 1990. №7. С.40–44.
- Евдокимов С.А., Рыбаков А.В., Соломенцев Ю.М.* Интегрированная интеллектуальная система ИнИС – оболочка для разработки и эксплуатации программных приложений пользователя // Информационные технологии. 1996. №3. С.10–13.
- Клименко Э.П., Селезнева А.П., Евдокимов С.А., Рыбаков А.В.* Система автоматизированной поддержки инженерных решений при проектировании электродвигателей // Автоматизация проектирования. 1998. №1. С.27–32.
- Краснов А.А., Пичугин В.И., Чередниченко Ю.В.* Проектирование штампов листовой штамповки при работе в САПИР // Там же. №3. С.13–17.
- Кун Т.* Структура научных революций. М., 1977.
- Кураксин С.А., Бикулов С.А., Баранов Л.В., Козлов С.Ю., Ксенофонтов Д.К., Ефремов А.Н.* T-FLEX CAD – новая технология построения САПР // Автоматизация проектирования. 1996. №1. С.50–54.
- Кузнечно-штамповочное производство. 2002. №5.
- Митрофанов С.П.* Групповая технология машиностроительного производства: В 2 т. Л.: Машиностроение, 1983.
- Пичугин В., Краснов А.* Автоматизация деятельности конструктора при проектировании штампов листовой штамповки // САПР и графика. 1999. №7. С.54–57.
- Рыбаков А.В.* Обзор существующих CAD/CAE/CAM-систем для решения задач компьютерной подготовки производства // Информационные технологии. 1997. №3. С.2–8.
- Рыбаков А.В.* Особенности выбора графической среды для промышленного проектирования объектов машиностроения // Там же. 2002. №5. С.13–20.
- Рыбаков А.В., Евдокимов С.А., Краснов А.А.* Проектирование технологической оснастки на основе системы автоматизированной поддержки информационных решений // Информационные системы. 2001. №10. С.15–22.
- Рыбаков А.В., Евдокимов С.А., Краснов А.А.* Создание систем автоматизации поддержки инженерных решений // Автоматизация проектирования. 1997. №5. С.44–51.

Рыбаков А., Евдокимов С., Краснов А., Никонов Н., Сабанин Д. Автоматизация деятельности конструктора технологической оснастки // САПР и графика. 2002. №8. С.74–78.

Рыбаков А.В., Евдокимов С. А., Мелешина Г.А. Создание автоматизированных систем в машиностроении. М.: Станкин, 2001.

Соломенцев Ю.М., Рыбаков А.В. Компьютерная подготовка производства // Автоматизация проектирования. 1997. №1. С.31–35.

Научное издание

*Соломенцев Юрий Михайлович
Митрофанов Владимир Георгиевич
Павлов Виктор Владимирович
Рыбаков Анатолий Викторович*

**Информационно-вычислительные системы
в машиностроении (CALS-технологии)**

Утверждена к печати
Ученым советом Института конструкторско-технологической
информатики РАН

Зав. редакцией Г.И.Чертова
Редактор В.С.Егорова
Художественный редактор Т.В.Болотина

Оригинал-макет подготовлен ООО «Янус-К».
109316, Москва, ул. Стройковская, д.12, корп.2.

Сдано в набор 25.12.2002. Подписано в печать 06.03.2003.
Формат 60х90/16. Бумага офсетная №1. Печать офсетная.
Уч.-изд л. 20. Физ.п.л. 18,25. Тираж 500. Заказ № 942.

Издательство «Наука». ЛР №020297 от 23.06.1997
117997 ГСП-7, Москва В-485, Профсоюзная ул., 90

E-mail: secret@naukaran.ru
Internet: www.naukaran.ru

Отпечатано в ФГУП «Производственно-издательский комбинат ВИНТИ»
140010, Люберцы, Октябрьский пр-кт, 403. т. 554-21-86