

УДК 004.891

А.С. Топал<sup>1</sup>, А.Н. Топал<sup>2</sup>, И.В. Шостак<sup>2</sup><sup>1</sup> Харьковская государственная академия дизайна и искусств, Харьков<sup>2</sup> Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков

## МЕТОДЫ И ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА СОЗДАНИЯ И ПОДДЕРЖКИ ЕДИНОГО ИНФОРМАЦИОННОГО ПРОСТРАНСТВА ТЕХНОЛОГИЧЕСКОЙ СЛУЖБЫ САМОЛЕТОСТРОИТЕЛЬНОГО ПРЕДПРИЯТИЯ

В статье рассмотрены проблемы комплексной автоматизации самолетостроительного предприятия путем синтеза мультиагентной CALS-системы на примере управления сборочным производством.

**Ключевые слова:** самолетостроение, комплексная автоматизация, CALS-система, сборочное производство, программный агент.

### Введение

Современная общемировая тенденция в организации производства на промышленных предприятиях, выпускающих сложную технику, заключается в комплексной их автоматизации на основе CAD/CAM/ CAE/PLM и CALS-систем [1]. Реализация такого подхода требует постоянной координации усилий большого числа специалистов, обеспечивающих различные этапы жизненного цикла изделий. При этом непременным условием эффективной координации работ специалистов при соблюдении единой стратегии является наличие гомогенной информационной среды в рамках отдельного предприятия или концерна. Такая среда призвана не только обеспечить автоматизацию выполнения отдельных подзадач, но и обеспечить обмен информацией между отдельными пользователями для координации их работы по поиску глобально оптимального или близкого к таковому решения ( поскольку локально оптимальное решение не учитывает общие для всей системы ограничения) [1]. В настоящий момент действует серия международных стандартов, представляющих CALS-технологии, среди которых требования по обеспечению единой информационной среды регламентирует стандарт ISO 10303 STEP (Standard for Exchange of Product data).

Попытки решения задачи эффективной организации командного взаимодействия в рамках единого информационного пространства предпринимаются различными учеными на протяжении последних 20 лет, при этом исследователи используют следующие методы и средства: технологии распределенного решения задач параллельного проектирования [2], общие архитектуры или платформы инженерных и промышленных приложений [1], мультиагентные системы [2], мультиэкспертные системы [3] и т.д.

Существующие в настоящее время CALS-системы [1] имеют ряд характерных особенностей. Они, как правило, разработаны по индивидуальному заказу под конкретную группу предприятий в рам-

ках одного концерна, а также ориентированы на крупносерийное и массовое производство.

**Цель статьи** состоит в описании программно-инструментальных средств комплексной автоматизации особого класса промышленных предприятий, характеризующихся единичным и мелкосерийным характером производства, частой сменой номенклатуры выпускаемых изделий и их модифицируемостью, а также большой долей ручного труда в производстве. Типичным примером такого предприятия является авиационный завод.

### 1. Характеристика проблемы

Из основных задач, на которые ориентированы CALS-системы [1], а именно структурирования и моделирования данных об изделиях и процессах, создания и сопровождения документации, необходимой для поддержки всех этапов жизненного цикла изделий, обеспечения эффективного управления и обмена данными между всеми участниками поддержки жизненного цикла изделий, наибольшие трудности вызывает реализация последней задачи. Данное обстоятельство обусловлено тем, что самолетостроительное предприятие является весьма сложным (в структурном, организационном и технологическом аспектах) объектом автоматизации. Исходя из особенностей рассматриваемого объекта [5], проблематика создания CALS-системы самолетостроительного предприятия лежит в аспектах разработки методического, организационного, математического и программного обеспечения CALS.

Таким образом, решение научной проблемы создания CALS-системы самолетостроительного предприятия состоит в последовательной реализации следующих этапов:

- обоснование выбора парадигмы, в рамках которой будет создаваться система (методический аспект разработки CALS);

- регламентация ролей и обязанностей участников жизненного цикла изделия (организационный аспект разработки CALS);

- модели и методы взаимодействия функциональных подсистем (математический аспект разработки CALS);

- разработка программно-инструментальных средств развертывания и поддержки единого информационного пространства (программный аспект разработки CALS).

## **2. Агентная парадигма как основа синтеза CALS-системы самолетостроительного предприятия**

Поскольку в комплексе работ по созданию такого сложного изделия как современный самолет, принимает участие большое количество различных служб, каждая из которых имея в своем составе множество структурных единиц, работает над решением своей локальной задачи, методы поддержки принятия координирующих решений в рамках одной службы (например, технологической) можно распространить и на все предприятие в целом.

Системы поддержки коллективной работы [4] делятся на два класса: мультиэкспертные системы (использующие концепцию «классной доски») и мультиагентные системы.

Под агентом мультиагентной системы принято понимать автономную программную сущность, которой делегирована часть полномочий пользователя в соответствии с ее функцией [4]. Каждый агент обладает определенной частью знаний об объекте проектирования и возможностью обмениваться этими знаниями с остальными агентами. В зависимости от типа, агент может поддерживать и интерфейс с пользователем. Под многоагентной системой будем понимать многокомпонентную систему, состоящую из агентов со специфицированным интерфейсом.

В мультиагентной системе каждый агент строит собственную модель текущего решения, основываясь на своих данных и данных других агентов. В таких системах имеются коммуникационный протокол и формат сообщений (язык коммуникаций), в соответствии с которым должны оформляться запросы и ответы [4]. Обобщенно можно сказать, что агенты автономны и гетерогенны, т.е. отсутствует единое управление. Коммуникации между агентами могут быть синхронными и асинхронными, направленными (peer-to-peer), общими (broadcast) или групповыми (multicast). Важно, что семантика сообщений между агентами должна быть высокого уровня. Это означает не тривиальную пересылку команд на запуск/останов, а полную реализацию информационных потоков между агентами в системе путем обмена сообщениями на языке, аналогичном языку высокого уровня в программировании [2].

Учитывая структуру технологической службы самолетостроительного предприятия (а именно

функциональную специализацию подразделений, а также деление их по видам выполняемых работ), а также изложенное выше описание подходов к организации коллективного взаимодействия можно сделать вывод о принципиальной возможности применения именно мультиагентной технологии при синтезе систем поддержки принятия координирующих решений на предприятии. В пользу этого также говорит постановка задач для подразделений технологической службы: расцеховка делает возможной параллельную работу подразделений и независимость на определенном этапе локальных решений между собой.

Использование технологии распределенных объектов на сегодняшний день может рассматриваться как необходимое, но, по-видимому, не достаточное условие для обеспечения единого информационного пространства (в том числе пространства знаний). С нашей точки зрения наиболее приемлемые решения лежат в использовании агентно-ориентированной архитектуры в качестве каркаса системы, в которой агенты обеспечивают динамическое связывание и взаимодействие, используя стандартные интерфейсы и форматы данных (или знаний). При этом вся бизнес-логика работы системы может обеспечиваться различными средствами – от простейших приложений до интеллектуальных систем поддержки принятия решений. Важен тот факт, что использование агентного подхода в таком случае сделает изначально закрытые системы открытыми и обеспечит возможность интеграции их в единое информационное пространство [4]. При этом также не будут потеряны ценные данные (знания), хранящиеся в этих системах.

Применение агентного подхода даст в результате возможность реализации CALS как единой системы с распределенным интеллектом, которая имеет множество положительных свойств: это и обработка информации, приближенная к местам ее получения, повышение надежности и отказоустойчивости системы в целом, эффективное распараллеливание работы различных структурных единиц предприятия, обеспечение полной наблюдаемости системы (что абсолютно невозможно в случае использования монолитной системы).

Поддержка моделей взаимодействия, основанных на концепции языка взаимодействия агентов (Knowledge Query and Manipulation Language - KQML), требует соответствующих соглашений между всеми подсистемами [1]. Поэтому первый вопрос, который возникает в процессе сопряжения различных систем или компонент, связан с введением в архитектуру дополнительного уровня или класса агентов, ответственного за преобразование форматов информационного обмена.

Для поддержки обработки описанного выше контекста сообщений между подсистемами могут

быть использованы специальные коммуникационные агенты (фасилитаторы, медиаторы или информационные агенты [3]), которые располагаются между отправителем и получателем сообщения и выполняют действия по стандартизации интерфейса, интеграции информации из различных источников, преобразованию запросов и ответов.

Использование фасилитаторов становится особенно важным в условиях интеграции новых информационных систем с существующими, которые не имеют необходимой степени гибкости в преобразовании входных/выходных форматов. Агенты (и другие программные компоненты системы) взаимодействуют с использованием фасилитаторов, которые могут выполнять преобразования специфических внутренних форматов представления данных (знаний) из/в стандартный формат обмена знаниями (Knowledge Interchange Format - KIF). Таким образом, каждый агент выполняет логический вывод в своих внутренних терминах, а получает и передает информацию другим агентам в необходимой им для понимания форме посредством фасилитаторов.

Каждый фасилитатор отвечает за обеспечение интерфейса между локальным объединением агентов (в рамках подсистемы) и удаленными агентами и преследует тем самым 4 основных цели: 1) обеспечивает надежный механизм передачи сообщений, 2) маршрутизирует сообщения по назначению, 3) преобразует приходящие сообщения в формат представления адреса, 4) осуществляет мониторинг агентов.

Таким образом, использование промежуточных агентов (фасилитаторов), особенно при условии интеграции существующих систем, позволяет получить гомогенное информационное пространство для работы всей системы, а также возможность отслеживания несоответствий при формировании решений в отдельных подсистемах и обеспечения их корректировки для поддержки адекватного формирования согласованных комплексных решений масштаба структурной единицы (или предприятия в целом). Результат будет зависеть от сложности исполнения фасилитаторов и от трактовки выполняемых ими задач (преобразование сообщений, корректировка результатов вывода подсистем и т.д.).

### **3. Архитектуры и спецификации агентных систем для поддержки единого информационного пространства в рамках технологической службы самолетостроительного предприятия**

Существующие варианты архитектур многоагентных систем и рациональный выбор архитектуры отдельного агента и многоагентной системы в целом существенно зависят концептуальной модели агента, принятого для ее описания формализма и языка спецификаций, математической модели коо-

перации агентов при совместном функционировании в системе, на какое приложение или класс приложений ориентирована многоагентная система, а также от ряда других факторов.

В настоящее время индустрия агентов регламентируется стандартами MASIF и FIPA, дающих рекомендации к созданию систем мобильных агентов и систем интеллектуальных агентов соответственно.

Серия стандартов FIPA описывает архитектуру мультиагентной системы и структуру самого агента, делая в основном упор на реализуемые агентами функции, и таким образом, могут являться основой для синтеза промышленной системы поддержки принятия координирующих решений на машиностроительном предприятии. Кроме того, стандартах FIPA описывается возможность интеграции агентной системы с существующим программным обеспечением, что является важным свойством создаваемой системы.

### **4. Реализация гибридного подхода к синтезу многоагентных систем на примере задачи актуализации текущего состояния производства в базах знаний и данных CALS**

Гибридный автоматный подход к синтезу мультиагентной CALS-системы подробно изложен в [5] и составляет основу ее математического обеспечения.

Рассмотрим особенности применения указанного подхода в приложении к задаче синтеза агента актуализации текущего состояния в базах знаний систем поддержки принятия решений сотрудников технологической службы самолетостроительного предприятия, а также кратко опишем созданный прототип системы.

Назначение системы актуализации текущего состояния: поддерживать актуальность данных и значений предикатов баз знаний систем поддержки принятия решений (СППР) представителей административно-управляющего персонала технологической службы, отражать в соответствующих подсистемах все изменения, происходящие в предметной области, получая актуальные данные от системы мониторинга, основанной на технологии PDA [6].

Входные данные: текущие (актуальные на определенный момент времени) значения переменных, определенных для предметной области в виде файла обмена данными определенного формата; правила отражения переменных в виде правил (занесены в базу правил агента).

Результаты работы агента: любые изменения текущего состояния объекта, получаемые от системы мониторинга или из других источников в виде изменения текущих значений переменных, непо-

средственно отражаются в подсистемах СППР, которые имеют возможность постоянно оперировать актуальными данными.

Режимы работы агента:

1. Наполнение базы правил и настройка параметров работы (в том числе подключение конкретных подсистем) – рис. 1.



Рис. 1. Интерфейс настройки агента мониторинга – режим выбора связи с базами данных обновления и выбора баз правил и файла изменений

2. Автономное функционирование в выбранном режиме (постоянно или по запросу).

На представленном рис. 1 показано:

– db\_equipment.mdb, dmetal.mdb, DB.mdb – базы данных, содержащие предикаты баз знаний соответствующих подсистем поддержки принятия решений (СППР по выбору оборудования, СППР по листовой и объемной штамповке и по наплавке);

– change.csv – файл с текущими значениями переменных объекта (получен от системы мониторинга);

– rules.mdb – база знаний в виде правил, которая используется для актуализации текущего состояния объекта мониторинга в базах знаний подсистем СППР.

Функционирование системы сводится к выполнению следующих действий:

1. Периодически или по запросу считывание файла с текущими данными из каталога обмена (определив предварительно наличие этого файла).

2. Синтаксический разбор считанного файла, удаление комментариев, возможных ошибок, считывание команд (пока что в стадии разработки), получение данных о текущем состоянии (формат согласовывается с системой мониторинга), запись их во внутренний формат агента.

3. Обработка каждой записи данных текущего состояния состоит в изменении предикатов баз знаний соответствующих подсистем и производится на основе правил, которыми заполняется база агента (таким образом реализуется полная независимость данных от кода и гибкость настройки агента под

конкретную базу знаний). В процессе вывода имеет-ся возможность отражать изменение одного параметра в нескольких базах знаний, а также отражать изменения сразу в нескольких параметрах в одной базе. Порядок действий агента на этом шаге:

– определение подсистемы, в которой производится обновление текущего состояния;

– поиск необходимой базы данных (как структурной составляющей базы знаний соответствующей подсистемы);

– идентификация изменяемого объекта в базе данных;

– выполнение необходимых действий над объектом (обновление, добавление, удаление);

– запись каждого обновления в общую базу данных текущего состояния с указанием даты и времени поступления данных (формирование базы данных текущего состояния с историей изменений);

– запись результатов действий в файл отчета (рис. 2);

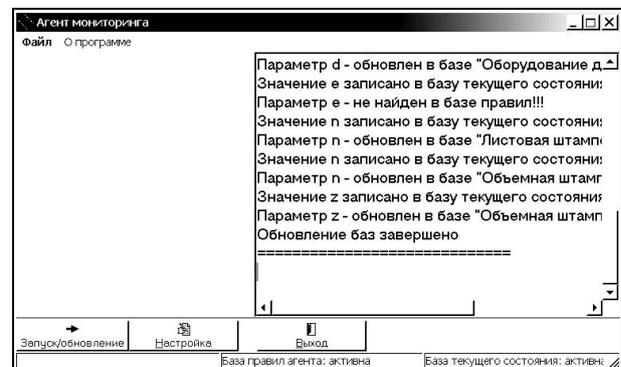


Рис. 2. Ведение протокола работы агента мониторинга

– в случае вызова по запросу программы, осуществляющей мониторинг, посылка этой программе сообщения о результатах обработки полученных данных.

Логически система отражения текущего состояния представляет собой несколько агентов (т.е. является по сути мультиагентной системой), каждый из которых обслуживает определенную подсистему принятия решений.

В целях удобства администрирования в прототипе системы конструктивно все агенты выполнены в виде потоков (threads) одного приложения, которое осуществляет все выше указанные действия. Существует возможность запуска одновременно нескольких процессов, осуществляющих обработку данных (при возросшей нагрузке на один модуль), а также разделение агента на несколько модулей, для обеспечения функционирования подсистем на физически различных платформах. Принципиально возможно получение данных от других источников (АРМ, система учета на складе и проч.), для этого необходима настройка базы правил агента и согла-

сование формата обмена данными. База правил агента структурирована относительно подсистем, поэтому обновление может происходить параллельно, не вызывая задержек и коллизий.

Программная реализация – подсистема актуализации данных мониторинга реализована с помощью интегрированной среды программирования Delphi 6.0 с применением возможностей многопоточной обработки данных. Так как агент функционирует автономно, следовательно сведен к минимуму его прикладной интерфейс, который позволяет настраивать основные параметры работы агента и просматривать совершенные им действия. Во время работы основное окно, а также окно настроек агента скрыто для пользователя, агент проявляет себя только значком в системном лотке Windows.

В результате актуализации баз знаний пользователь получает новое качество исходной системы: постоянно обновляемую базу знаний, которая может осуществлять вывод на знаниях, основываясь на текущем состоянии ресурсов.

При разработке системы также была проведена реализация прототипа подобной системы на основе существующего стандарта ДСТУ 4071–2001 на использование технологии CORBA для построения распределенных систем.

### **5. Разработка типового фрагмента мультиагентной CALS-системы по технологической подготовке сборочного производства самолетостроительного предприятия**

На основании обзора агентных систем был выбран стандарт FIPA на создание программных агентов, т.к. именно этот стандарт полностью реализует идеи интеграции создаваемой СППР в существующую систему (в данном случае АСУ).

На рис. 3 применительно к разрабатываемой СППР коммуникационный канал реализован в виде протокола обмена сообщениями между агентами, агент-брокер используется только при реализации СППР с использованием технологии CORBA, агент-обложка реализован в виде модуля, вызывающего функции программного интерфейса существующего программного обеспечения (например, API САПР ТП «Компас-Вертикаль»), агент-клиент представляет собой типовой блок разрабатываемой СППР (т.е. координатора и распознавателя, связанных между собой).

После выбора принципиальной агентной платформы необходимо синтезировать типовую структуру блока принятия решений, описываемого гибридным автоматом. Такая структура представлена ниже на рис. 3. Для общения между агентами может быть использован стандартный агентный «язык» KQML [7], но по причине необходимости передачи

информации лишь в форме предикатов, для конкретной реализации был выбран формат обмена на основе XML. С его помощью была реализована передача предикатов в создаваемой системе (например, для обмена информацией между координатором и распознавателем).

После представления общей структуры решающего блока (агента) была синтезирована структуру СППР в целом (или типового ее фрагмента). Эти структуры изображены на рис. 4.

Интеграция СППР показана на примере пакета «Компас-Автопроект», который чаще всего используется на отечественных самолетостроительных предприятиях и имеет развитый программный интерфейс, позволяющий получать всю необходимую информацию как о технологическом процессе сборки или изготовления детали (вплоть до отдельных переходов), так и о параметрах каждой операции таких технологических процессов. Программный комплекс КОМПАС-Автопроект состоит из двух взаимосвязанных подсистем: КОМПАС-Автопроект-Спецификации и КОМПАС-Автопроект-Технология. В первой из названных подсистем концентрируются технологические задачи, связанные с составом изделия, а во второй – с проектированием технологических процессов. Такая схема построения продиктована необходимостью интеграции технологического модуля с PDM-системами – как собственными, так и внешними.

На рис. 5 изображена диаграмма развертывания СППР и интеграция ее с существующими компонентами информационной системы цехового уровня.

### **6. Оценка качества формируемых решений по сборке самолетных конструкций**

Качество координирующих решений, формируемых прототипом СППР, оценивалось на примере реальных данных (типовые ситуации появления брака во время внестapelной сборки в цехе №27), из которых формировались сценарные примеры для обучения и проверки системы (правила для обучающей и предикаты для контрольной выборки), а также было проведено сравнение с результатами решения этих ситуаций несколькими экспертами.

Качество определялось общей ошибкой принятия неверного решения, получаемой путем взвешенного усреднения ошибок первого рода (нераспознаваемых ситуаций) и второго рода (ситуаций, классифицированных как коллизии, но не являющихся таковыми).

Для оценки качества работы системы применялся метод скользящего контроля [6], а именно разбиение с одним отделяемым объектом (LOOCV – leave one out cross validation).

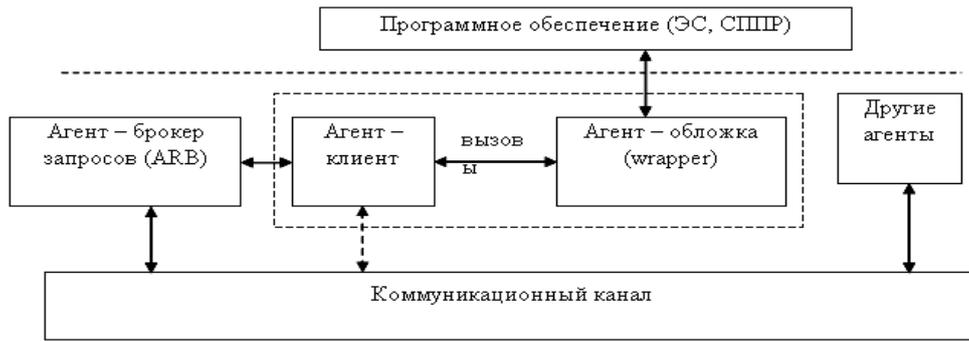


Рис. 3. Схема интеграции агента с существующим программным обеспечением (FIPA '2000)

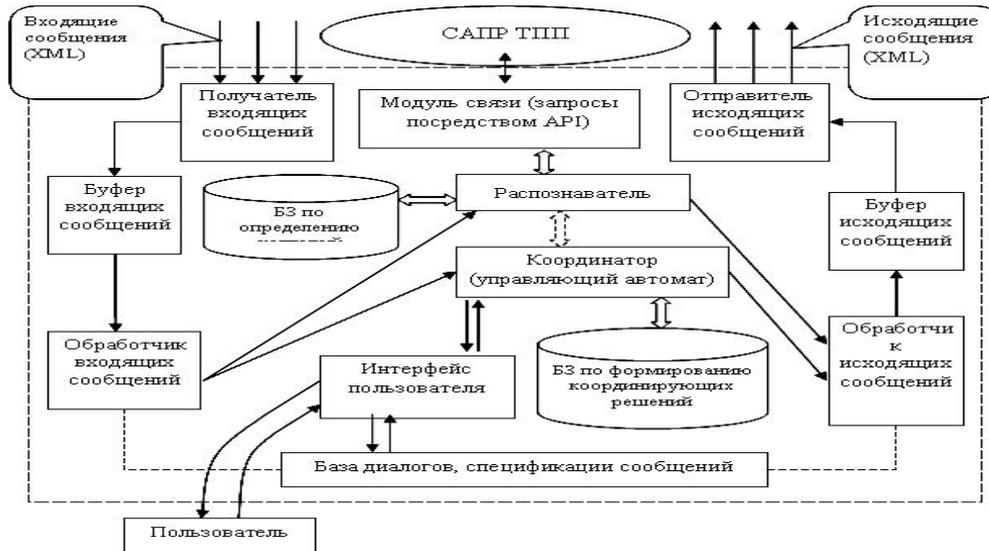


Рис. 4. Функциональная схема автоматной реализации СППР по формированию координирующих решений в сборочном производстве самолетостроительного предприятия

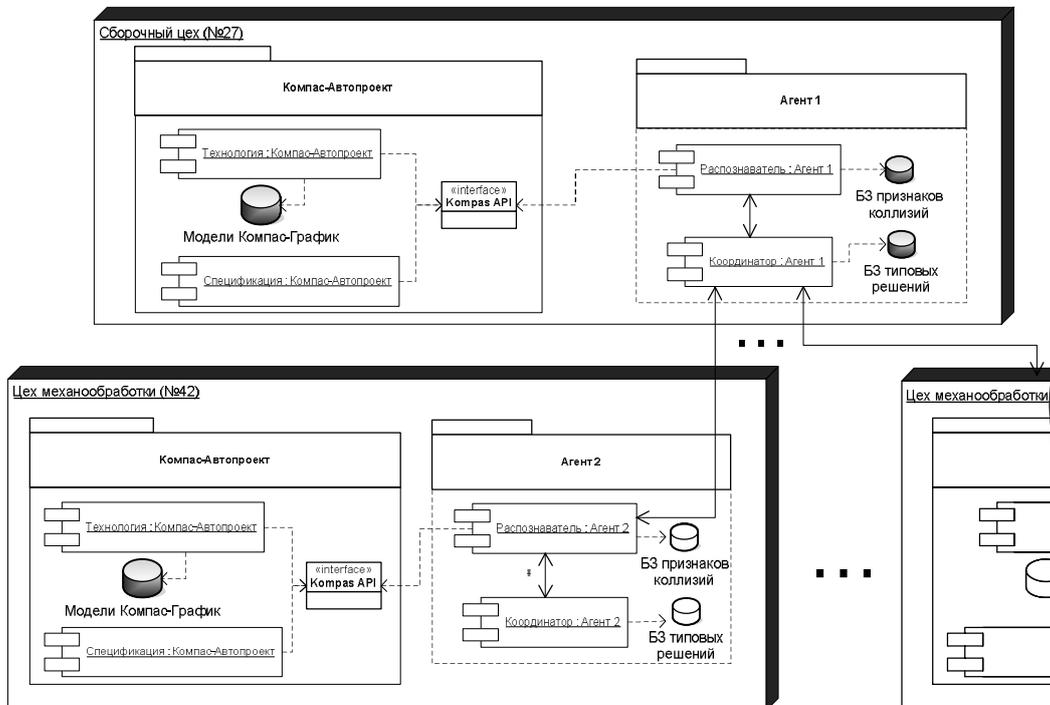


Рис. 5. Место СППР по формированию координирующих решений в общей структуре АСУ уровня цеха сборки самолетных конструкций Харьковского государственного авиационного производственного предприятия (на примере интеграции с системой «Компас-Автопроект»)

Метод скользящего контроля такого вида (в противоположность методу «5-2» или k-кратному разбиению) был выбран во избежание попадания нескольких схожих обучающих примеров в одну часть разбиения, тем самым возможно получение смещенной оценки ошибки.

Особенностью выбранного метода состоит в необходимости раздельного учета ошибок первого и второго рода. Анализ статистики затрат при возникновении ошибок первого и второго рода (по обучающей выборке) показывает, что затраты на исправление неправильного решения для ошибок первого и второго рода приблизительно равны  $P(y = \pm 1 = 73\%$  и  $P(y = -1 = 27\%$  соответственно (если взять за 100% сумму этих затрат по одной и той же ситуации), т.е. убытки от нераспознанной коллизии намного больше, чем убытки от «ложной» коллизии (при условии что решение по ней будет принято, т.е. будет изменен технологический процесс изготовления сборочной единицы и потребуются доработка детали впоследствии).

Так как ошибки неравнозначны, то для оценки общей ошибки системы необходимо брать взвешенные значения ошибок первого и второго рода с весами соответственно 0,73 и 0,27 (несмотря на то, что ошибочно распознанная коллизия может быть легко проигнорирована пользователем при составлении результирующего документа).

Для тестирования была составлена выборка из 30 наиболее типовых прецедентов, имевших место в статистике самолетостроительного предприятия за определенный период. С привлечением экспертов для каждого случая были выделены обучающие правила, которые использовались при моделировании, а также предикаты, которые использовались в качестве исходных при проверке на каждом шаге разбиения.

Для сравнения результатов моделирования указанные ситуации были предложены двум экспертам (ведущим технологам) для решения. Сводные результаты представлены в табл. 1 (взвешенная ошибка вычислялась с использованием ранее указанных весов ошибки I и II рода).

Можно видеть, что имеющиеся в прототипе СППР правила позволяют формировать решения, вероятность ошибки в которых находится на уровне 16%, что приблизительно соответствует значению ошибки эксперта. Можно также сделать вывод, что качество работы системы определяется качеством обучения базы знаний, которую она использует для распознавания и решений потенциальных коллизий.

Таким образом, увеличивая размер выборки, или проводя извлечения знаний экспертов, можно добиться улучшения качества работы системы.

Таблица 1

Результаты моделирования процесса формирования координирующих решений

	эксперт1	эксперт2	СППР	эксперт1, %	эксперт2, %	СППР, %
Пропущенные коллизии (ошибка I рода)	7	3	4	23,33%	10,00%	13,33%
Излишние коллизии (ошибка II рода)	4	3	7	13,33%	10,00%	23,33%
Взвешенная ошибка				20,63%	10,00%	16,03%

Главное же преимущество СППР состоит в том, что эффективность ее работы постоянна, в то время как человеку свойственно увеличение ошибок из-за большого количества ситуаций, что имеет место на реальном предприятии. По имеющейся статистике затрат на исправление брака были выделены виды брака, которые определяются коллизиями, распознаваемыми в автоматизированном режиме в СППР (в данной тестовой выборке это ситуации монтажа замков, установки крепежа, установки готовых изделий, совместной клепки шпангоутов и элементов интерьера). Уровень затрат на исправление брака относительно прямых материальных затрат, а также соотношение затрат на исправление брака по видам представлено на рис. 6.

Анализируя второй график, можно сказать, что среднее значение затрат на исправление брака по ситуациям, которые распознаются системой, составляет около 44%. Следовательно, учитывая общую

ошибку при работе системы, как вероятность получить неверное решение, равную 0,16, можно прогнозировать возможность уменьшения затрат, связанных с исправлением брака при сборке в среднем на  $44\% * 0,16 = 37\%$ .

Следует заметить, что непроизводительные расходы на исправление брака на самолетостроительных предприятиях достигают до 15-20%, особенно при запуске нового изделия в серию или при проведении масштабных модификаций.

Так по статистике на первой диаграмме (рис. 6.) среднее значение затрат на исправление брака в сборочном цехе составило около 7,2%. Следовательно, можно сделать вывод, что в результате применения разработанных моделей и методов (с учетом качества координирующих решений, формируемых системой) можно было ожидать снижения непроизводительных затрат по цеху в среднем на 2,7% (относительно прямых материальных затрат).

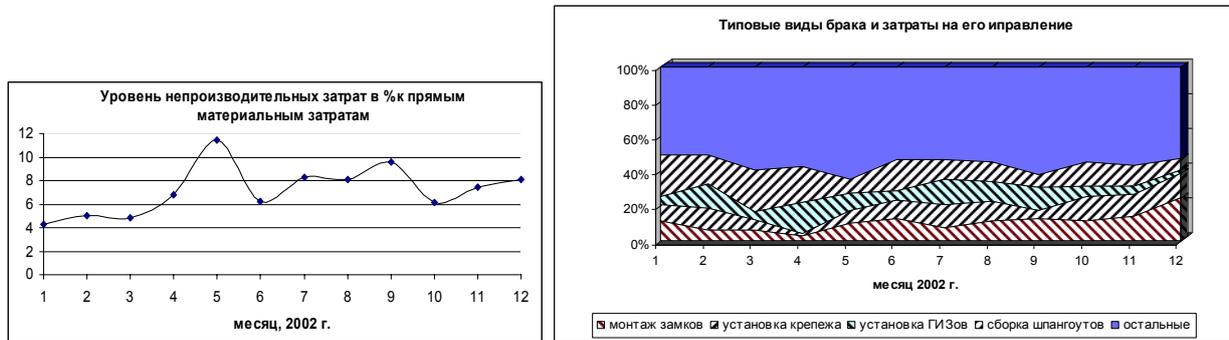


Рис. 6. Затраты на исправление брака и их качественный состав (по ситуациям, распознаваемым в СППР)

**Выводы**

В статье проведен анализ возможностей применения существующих методов и средств построения информационных систем класса CALS для обеспечения принятия координирующих решений по технологической подготовке сборочного производства на самолетостроительном предприятии. В качестве основы для построения подобной системы была выбрана многоагентная парадигма, в рамках которой осуществлена программная реализация типового фрагмента СППР в виде интеллектуального агента.

Гибридный подход к синтезу многоагентных систем был продемонстрирован с помощью прототипа агента актуализации текущего состояния производства в базах знаний и данных СППР.

С помощью разработанного прототипа СППР при разработке сборочных технологических процессов была проведена оценка эффективности разработанного метода формирования координирующих решений. Оценка проводилась с помощью метода скользящего контроля с одним отделяемым объектом. (LOOCV). В результате моделирования был получен уровень общей ошибки около 16%, что позволит в среднем снизить затраты на исправления брака на 37% (согласно статистическим данным объемов брака и затрат на его исправление по конкретным видам разрешимых коллизий).

**Список литературы**

1. Норенков И.П. Информационная поддержка наукоемких изделий. CALS-технологии / И.П. Норенков, П.К. Кузьмик. – М.: МГТУ им. Н.Э. Баумана, 2002. – 320 с.
2. Алиев Р.А. Методы интеграции в системах управления производством / Р.А. Алиев. – М.: Энергоатомиздат, 1989. – 269 с.
3. Алиев Р.А. Методы и алгоритмы координации в промышленных системах управления / Р.А. Алиев, М.И. Либерзон. – М.: Радио и связь, 1987. – 208 с.
4. Шостак И.В. Интеллектуализация процессов технологической подготовки производства на основе мультиагентной технологии / И.В. Шостак, А.С. Топал // Вестник двигателестроения. – 2003. – № 2. – С. 187-191.
5. Шостак И.В. Проблемы разработки мультиагентной интеллектуальной интегрированной системы поддержки принятия решений в авиационном производстве / И.В. Шостак, Л.А. Гордиенко, Е.П. Киричук, А.С. Топал // Авиационно-космическая техника и технология. – 2003. – № 8 (43). – С. 14-22.
6. Топал А.С. Формирование комплексных решений в интеллектуальных производственных системах / А.С. Топал // Открытые информационные и компьютерные интегрированные технологии. – X: Нац. аэрокосм. ун-т «ХАИ», 2005. – Вып. 26. – С. 106-110.

Поступила в редколлегию 3.10.2011

**Рецензент:** д-р техн. наук, проф. В.М. Левыкин, Харьковский национальный университет радиоэлектроники, Харьков.

**МЕТОДИ І ІНСТРУМЕНТАЛЬНІ ЗАСОБИ СТВОРЕННЯ І ПІДТРИМКИ ЄДИНОГО ІНФОРМАЦІЙНОГО ПРОСТОРУ ТЕХНОЛОГІЧНОЇ СЛУЖБИ ЛІТАКОБУДІВНОГО ПІДПРИЄМСТВА**

О.С. Топал, О.М. Топал, І.В. Шостак

У статті розглянуті проблеми комплексної автоматизації літакобудівного підприємства шляхом синтезу мультиагентної CALS-системи на прикладі управління збиральним виробництвом.

**Ключові слова:** літакобудування, комплексна автоматизація, CALS-система, збиральне виробництво, програмний агент.

**METHODS AND TOOLS OF CREATION AND SUPPORT OF SINGLE INFORMATIVE SPACE OF TECHNOLOGICAL SERVICE OF aircraft construction ENTERPRISE**

A.S. Topal, A.N. Topal, I.V. Shostak

This article is dedicated to the problem of composite automatization in aircraft by synthesis multiagent CALS-system for example of assembling processes control.

**Keywords:** aircraft construction, complex automation, CALS-system, frame-clamping production, programmatic agent.