

2. Система тестирования включает вопросы по технологическим процессам, взаимодействует с базами данных локального и сетевого назначения и обеспечивает функциональную взаимосвязь с математической моделью производства линейных алкилбензолов, что позволяет рассчитывать оптимальную подачу диметилдисульфида в реактор гидрирования.
3. Программно реализована фреймовая модель анализа и устранения аварийных ситуаций. С ее помощью путем обучения инженерно-технического персонала будет обеспечен высокий уровень безопасности эксплуатации установок производства.
Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (11-07-98001-р-сибирь-а, Региональный конкурс СИБИРЬ).

СПИСОК ЛИТЕРАТУРЫ

1. Кравцов А.В., Ивашкина Е.Н., Юрьев Е.М., Иванчина Э.Д. IT-технологии в решении проблем промышленного процесса дегидрирования высших парафинов. — Томск: STT, 2008. — 230 с.
2. Иванчина Э.Д., Ивашкина Е.Н., Шарова Е.С. Системный анализ химико-технологических процессов. — Томск: Изд-во ТПУ, 2008. — 95 с.
3. Архангельский А.Я. Программирование в Delphi 7. — М.: ООО «Бином-Пресс», 2003. — 1152 с.
4. Сорокин А.В. Delphi. Разработка баз данных. — СПб.: Питер, 2005. — 477 с.
5. Кравцов А.В., Иванчина Э.Д. Интеллектуальные системы в химической технологии и инженерном образовании: Нефтехимические процессы на Pt-катализаторах. — Новосибирск: Наука, 1996. — 200 с.
6. Долганов И.М., Францина Е.В., Афанасьева Ю.И., Иванчина Э.Д., Кравцов А.В. Моделирование промышленных нефтехимических процессов с использованием объектно-ориентированного языка Delphi // Известия Томского политехнического университета. — 2010. — Т. 317. — № 5. — С. 53–57.

УДК 004.422.8:004.043

ИСПОЛЬЗОВАНИЕ УПРАВЛЯЕМОГО ПРОГРАММНОГО КОДА НА ПЛАТФОРМЕ.NET FRAMEWORK ПРИ РАЗРАБОТКЕ АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ АНАЛИЗА И ПРОГНОЗИРОВАНИЯ РАБОТЫ ДОМЕННОГО ЦЕХА

В.В. Лавров, Н.А. Спирин, А.А. Бурыкин, В.Ю. Рыболовлев*, А.В. Краснобаев*

ФГАОУ ВПО «Уральский федеральный университет имени первого Президента России Б.Н. Ельцина», г. Екатеринбург

*ОАО «Магнитогорский металлургический комбинат», г. Магнитогорск

E-mail: lv@tim.ustu.ru

Отражены особенности применения управляемого программного кода при разработке современных информационных систем на платформе .NET Framework (CLR). Представлены варианты построения архитектур автоматизированной информационной системы анализа и прогнозирования производственных ситуаций доменного цеха ОАО «Магнитогорский металлургический комбинат».

Ключевые слова:

Технология разработки программного обеспечения, Common Language Runtime, .NET Framework, SQL Server, системы поддержки принятия решений, доменное производство, технологические задачи в металлургии.

Key words:

Software engineering, Common Language Runtime, .NET Framework, SQL Server, decision support systems, blast furnaces, technological problems in metallurgy.

Создание и успешное внедрение в практику работы металлургических предприятий современных автоматизированных информационных систем во многом зависит от используемых технологий разработки и выбора инструментальных средств их программной реализации.

В доменном цехе ОАО «Магнитогорский металлургический комбинат» разработана и проходит опытно-промышленные испытания автоматизированная информационная система анализа и прог-

нозирования производственных ситуаций (АИС АППС), которая помогает инженерно-технологическому персоналу решать комплекс технологических задач, направленных, в конечном итоге, на повышение эффективности доменной плавки [1, 2].

Разработка системы проводилась поэтапно в течение длительного времени, при этом были рассмотрены и частично реализованы несколько архитектур ее построения. Упрощенная архитектура

ра первоначальной версии АИС АППС представлена на рис. 1, где отражены ее основные подсистемы.

1. *Подсистема визуализации показателей работы доменных печей.* Основной функцией подсистемы является предоставление пользователю возможностей отображения в удобном виде показателей работы доменных печей за определенный период (смена, сутки), формирования отчетов и экспорта численных данных в стандартном формате для дальнейшего анализа программными средствами. Визуализация данных производится в двух режимах: динамика изменения данных по всем доступным показателям для отдельной доменной печи; динамика изменения величины по одному выбранному показателю для всех доменных печей.

2. *Подсистема формирования технического отчета о работе доменных печей и цеха.* Подсистема служит для предоставления пользователю возможности автоматизированного ввода, хранения, обработки, манипулирования и представления в удобном виде всех отчетных данных о: состоянии печей доменного цеха; количестве и качестве полученного чугуна; выполнении графиков выпусков; расхо-

де, остатках, химическом составе сырья; качестве кокса и др. В подсистеме предусмотрены автоматизированный сбор и подготовка необходимых отчетных данных о работе доменного цеха за нормативный период (месяц), а также произвольно указанный пользователем период работы (оперативный период). В качестве источников данных могут выступать не только отчетные данные в виде твердых копий документов, но и результаты работы АРМов доменных печей, централизованные информационные ресурсы комбината (корпоративная информационная система, центральная диспетчерская комбината и др.).

3. *Подсистема сопоставительного анализа работы доменных печей и цеха.* Обеспечивает возможность сравнения работы цеха или печи по комплексу выбранных параметров на основе автоматического извлечения отчетных показателей из базы данных Информационно-вычислительного центра доменного производства и сравнения результатов работы печей и цеха по фактическим данным за два произвольно выбранных периода работы цеха. Содержанием данных являются разделы технического отчета доменного цеха.

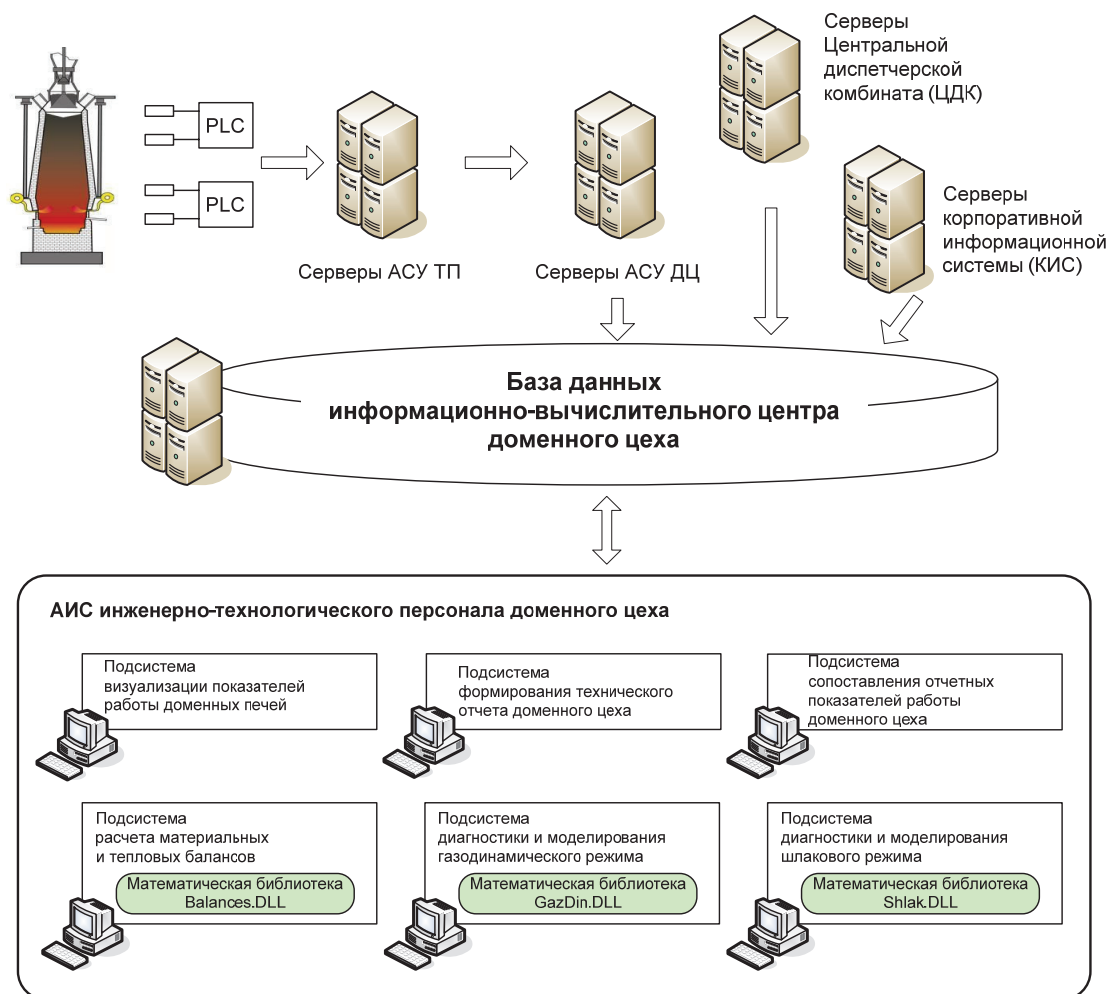


Рис. 1. Архитектура АИС инженерно-технологического персонала доменного цеха с использованием внешних математических библиотек

Остальные подсистемы (*расчета материальных и тепловых балансов доменных печей, диагностики и моделирования газодинамического и шлакового режимов доменной плавки*) построены на основе алгоритмов расчета с использованием математических моделей доменного процесса для выбранного пользователем периода работы отдельных печей или цеха в целом. Реализация математических алгоритмов расчета выполнена в виде программных библиотек динамической компоновки (DLL-библиотек) для каждой из подсистем.

Динамическая компоновка представляет способ вызова функции, который не является частью исполняемого кода приложения. Исполняемый код функции расположен в библиотеке DLL, которая содержит несколько скомпилированных, связанных и отдельно сохраненных функций в используемых процессах. Динамическая компоновка позволяет исполняемым модулям, например, ехе-файлу, включать только необходимую информацию в среду выполнения и размещать исполняемый код в функции DLL.

Механизм DLL-библиотек появился вместе с операционной системой Windows и является ее неотъемлемой частью. Суть этого механизма в том, что в процессе компоновки исполняемого модуля приложения с использованием внешних процедур (например, математических алгоритмов расчета) в него помещаются не сами процедуры, а только их названия (номера) вместе с названиями DLL-библиотек, в которых они содержатся. Динамическая компоновка имеет следующие основные преимущества:

- экономия оперативной памяти и дискового пространства. Одну библиотеку DLL могут использовать одновременно несколько программных процессов, при этом в оперативной памяти хранится только один ее экземпляр для всех процессов. Кроме этого, многие приложения совместно используют одну копию библиотеки DLL на диске;
- простота внесения обновлений в библиотеку DLL. При изменении функций в библиотеке DLL не требуется перекомпиляция или переконкампонка приложений, использующих эти библиотеки, если аргументы функций и возвращаемые значения остались прежними.

Однако за десятилетия использования DLL-библиотек программисты столкнулись с существенными проблемами, которые получили обобщенное название «DLL hell» [3]. Суть проблемы сводится к следующему – поддержка обратной совместимости DLL-библиотек и взаимозаменяемость в обе стороны никак не регламентирована, несовместимые DLL легко замещают друг друга или отключают друг друга. Отсутствие стандарта на процедуру установки приводит к тому, что установка новых программ приводит к замещению работающих DLL на несовместимые версии. Поскольку каждый программист решает эти задачи по-своему, в конечном итоге это приводит к полнейшей не-

разберихе и вызывает существенные сложности в поддержке программных продуктов. Эта проблема явилась одним из основных факторов создания платформы .NET Framework.

Появление и широкое распространение современной платформы создания, развертывания и исполнения распределенных приложений Microsoft .NET Framework позволяет использовать в практике разработки информационных систем преимущества общезыковой исполняющей среды (Common Language Runtime, CLR). Программный код, исполняемый под управлением CLR, называют управляемым кодом. Microsoft .NET Framework включает в себя средства организации взаимодействия с операционной системой и прикладными программами, методы доступа к базам данных, средства поддержки распределенных (сетевых) приложений, языки программирования, унифицированные интерфейсы пользователя.

Начиная с версии SQL Server 2005, сервер базы данных обеспечивает интеграцию с компонентами CLR платформы .NET Framework для Microsoft Windows [4]. Это означает, что для создания объектов базы данных и извлечения и обновления данных в базах данных Microsoft SQL Server помимо традиционного языка программирования Transact-SQL можно использовать языки .NET Framework. В частности, на Microsoft Visual C# можно создать хранимые процедуры, триггеры, статистические выражения, пользовательские функции и пользовательские типы

С учетом вышесказанного разработана усовершенствованная архитектура АИС АППС, которая в настоящее время реализована и внедрена в доменном цехе ОАО «ММК» (рис. 2).

Как видно из рис. 2, все математические библиотеки информационно-моделирующих подсистем реализованы в виде программных сборок (assembly), которые зарегистрированы на сервере базы данных Microsoft SQL Server и осуществляют доступ к внешним хранимым процедурам CLR.

Для примера приведем фрагмент листинга Transact-SQL создания внешней хранимой процедуры EXTDLL_Balances для подключения модуля расчета теплового баланса из управляемого кода (сборки) Balances.DLL.

```
CREATE PROCEDURE [dbo].[EXTDLL_Balances]
  (@dtFirstDay [nvarchar](1024), @NPech [int])
AS EXTERNAL NAME Balances.StoredProcedures.BalanceTeplo
```

В качестве параметров процедуры указаны два: @dtFirstDay (первый день отчетного месяца) и @NPech (номер печи). Параметры, передаваемые в хранимую процедуру CLR, могут быть любого собственного типа Microsoft SQL Server, имеющего эквивалент в управляемом коде. Параметры, возвращающие табличные значения, передаются в процедуру или функцию CLR, предоставляя эффективный способ передачи на сервер нескольких

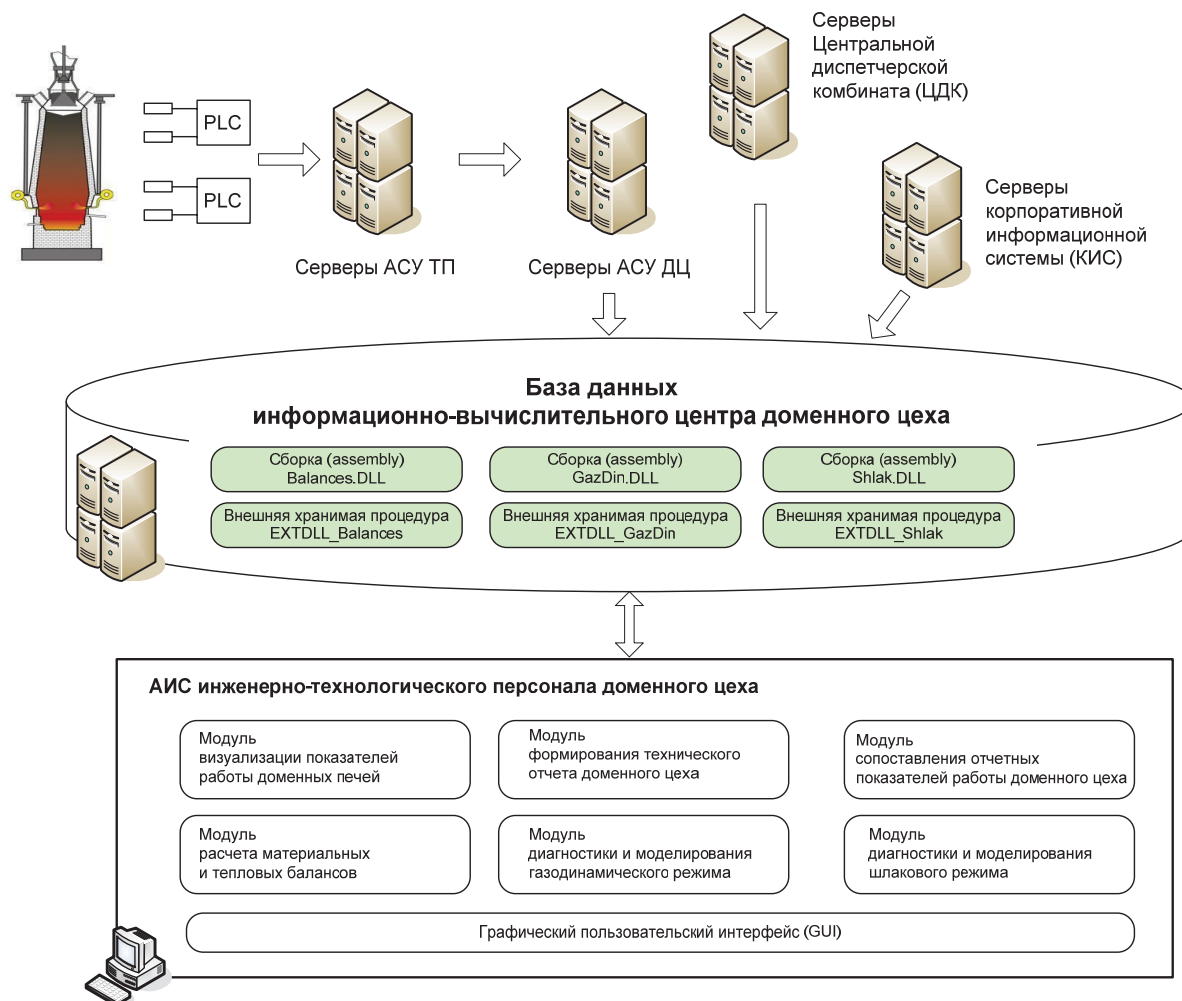


Рис. 2. Архитектура АИС инженерно-технологического персонала доменного цеха с использованием объектов управляемого кода базы данных

строк данных. Возвращающие табличное значение параметры выполняют функции, аналогичные массивам параметров, но обладают большей гибкостью и лучше интегрируются с Transact-SQL. Они также обеспечивают возможность повышения производительности. Кроме того, возвращающие табличное значение параметры способствуют сокращению циклов приема-передачи данных с сервера и на сервер.

На рис. 3 представлена диаграмма взаимодействия объектов в ходе работы программного модуля диагностики и моделирования газодинамики доменных печей. Пользователь работает с системой через клиентское приложение с двумя основными формами: «Диагностика газодинамики в базовом периоде» и «Оценка газодинамики в проектном периоде». Взаимодействие клиентского приложения с сервером базы данных осуществляется только с хранимой процедурой GazDin, все остальные операции по извлечению данных, расчету газодинамических показателей и др. скрыты от приложения и производятся сервером базы данных в фоновом режиме. Хранимая процедура GazDin

имеет два входных параметра: @dtFirstDay и @IsBase (логическая переменная, отражающая необходимость расчета газодинамики в базовом или проектном периодах). Расчет газодинамических показателей по математической модели производится путем обращения в сборку GazDin.DLL, написанной на Microsoft Visual C# и расположенной на сервере базы данных Report в ядре Microsoft SQL Server.

Среди основных преимуществ построения АИС АППС на основе реализации объектов управляемого кода базы данных на языке.NET Framework вместо Transact-SQL отметим следующие:

- *Расширенная модель программирования.* Языки.NET Framework предлагают конструкции и возможности, ранее недоступные для разработчиков Microsoft SQL Server в Transact-SQL, например выполнение математических вычислений с использованием сложных формул, новые пользовательские типы и статистические выражения, создание массивов, обработка сложных исключений и повторное использование кода, расширяющие возможности Microsoft SQL Server по созданию запросов;

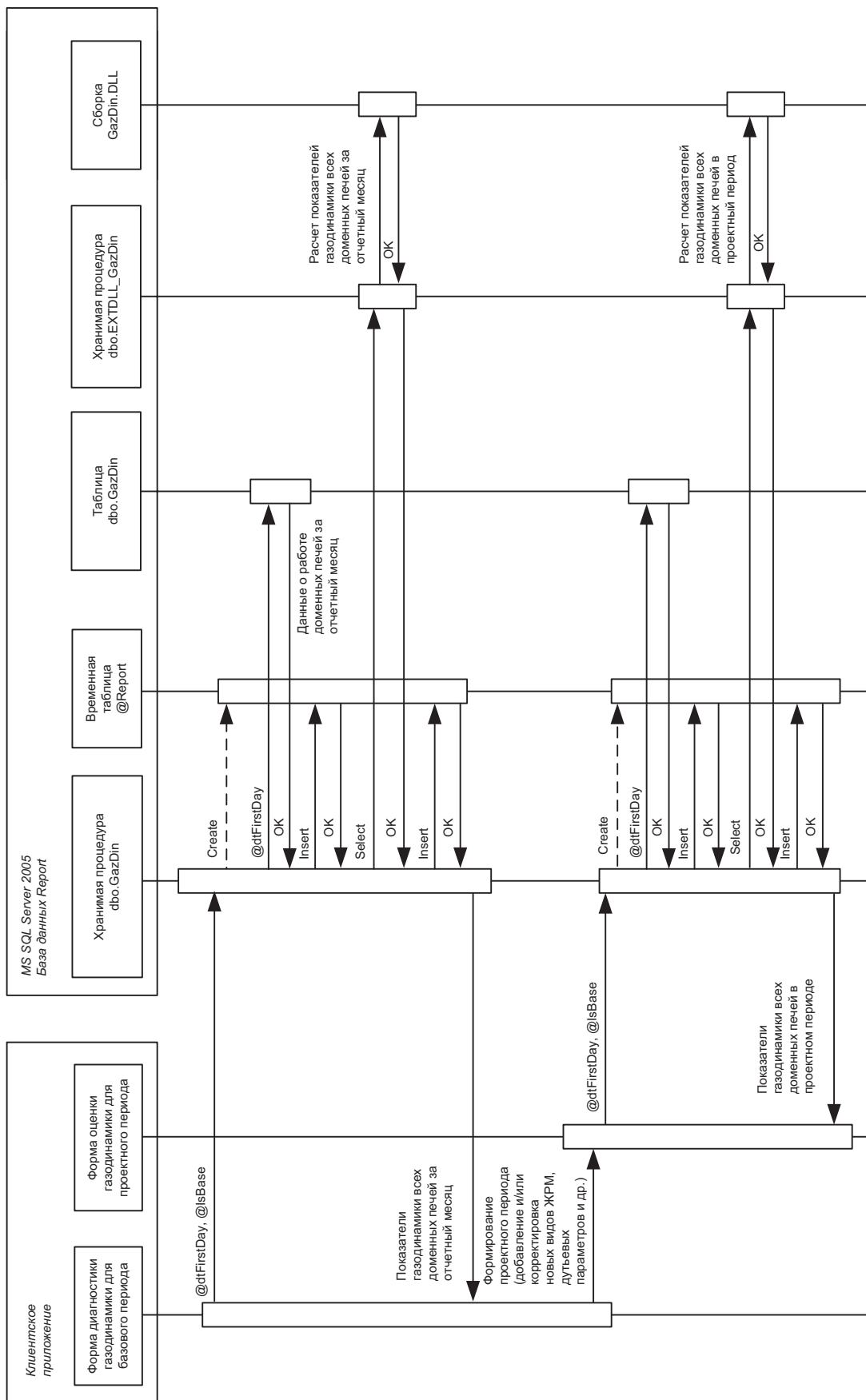


Рис. 3. Диаграмма взаимодействия объектов в ходе расчета газодинамики доменных печей в базовом и проектном периодах

- *Улучшенная стабильность и повышенная надежность.* Объекты базы данных, создаваемые с помощью Microsoft Visual C#, являются более безопасными, стабильными и надежными, чем расширенные хранимые процедуры, которые могут вызывать утечки памяти или другие проблемы, снижающие производительность и надежность сервера базы данных. При запуске хранимых процедур, созданных в Microsoft Visual C#, управление памятью и работа с потоками хранимой процедурой не выполняются и, таким образом, обрабатываются более надежно;
- *Улучшенная безопасность.* Управляемый код выполняется в общезыковой среде выполнения CLR, размещенной в ядре базы данных. Это делает объекты базы данных платформы .NET Framework более безопасными по сравнению с расширенными хранимыми процедурами в более ранних версиях Microsoft SQL Server. При использовании объектов базы данных, созданных в Microsoft Visual C#, управление доступом для кода в этих языках сочетается с пользовательскими разрешениями в Microsoft SQL Server;
- *Общая среда разработки систем.* Разработка баз данных полностью интегрирована в среду разработки Microsoft Visual Studio. Для создания и отладки объектов баз данных и скриптов использованы те же средства, что и для написания компонентов и служб платформы .NET Framework промежуточного или клиентского уровня;
- *Повышенная производительность.* Отдельные функции, например выполняющие математические операции над каждой строкой в базе данных, будут выполняться эффективнее за счет того, что они представляют собой скомпилированные сборки, построенные на языке Microsoft Visual C#, в отличие от традиционного языка базы данных Transact-SQL, который представляет собой интерпретируемый код. Разработанная система полностью интегрирована в информационную структуру предприятия, в процессе эксплуатации взаимодействует с другими программно-аппаратными информационными

системами комбината. Обеспечение отдельных функций системы производится на основе автоматического выполнения задач по расписанию, например функции сбора производственных данных и подготовки отчетных документов за истекший отчетный период [5].

Опыт эксплуатации АИС АППС показал, что ее применение в доменном производстве имеет несомненные преимущества и приносит положительный эффект. Это обусловлено повышенными рисками от принятия необоснованных, неправильных управленческих решений с учетом относительно больших масштабов потребления сырьевых и топливно-энергетических ресурсов, а также выпуска продукции доменного цеха. Инженерно-технологический персонал доменного цеха с помощью программного комплекса имеет возможность оперативно производить анализ работы металлургических агрегатов, оценивать текущее состояние хода технологического процесса, производить изучение и анализ наметившихся отклонений ключевых показателей, выявлять причины, повлекшие эти отклонения и разрабатывать мероприятия по повышению эффективности металлургического производства.

Выводы

В доменном цехе ОАО «Магнитогорский металлургический комбинат» реализована усовершенствованная автоматизированная информационная система анализа и прогнозирования работы доменного цеха, включающая базу данных Microsoft SQL Server и клиентские приложения информационно-моделирующих подсистем на платформе .Net Framework. Показаны преимущества использования управляемого программного кода (CLR) при создании математических библиотек информационно-моделирующих подсистем в виде программныхборок (assembly) на сервере базы данных Microsoft SQL Server. Применение предложенного подхода позволило снизить трудозатраты на реализацию и сопровождение компьютерной системы поддержки принятия решений за счет повышения надежности и стабильности работы отдельных подсистем.

СПИСОК ЛИТЕРАТУРЫ

1. Спирин Н.А., Лавров В.В., Рыболовлев В.Ю., Краснобаев А.В., Оморин О.П., Косаченко И.Е. Модельные системы поддержки принятия решений в АСУ ТП доменной плавки / под ред. Н.А. Спирина. – Екатеринбург: УрФУ, 2011. – 462 с.
2. Лавров В.В., Спирин Н.А., Бурыкин А.А., Краснобаев А.В. Создание программного комплекса «АРМ технолога доменного цеха» на основе современных информационных технологий // Сталь. – 2010. – № 1. – С. 17–21.
3. DLL hell / Материал из Википедии – свободной энциклопедии. 2011. URL: http://ru.wikipedia.org/wiki/DLL_hell (дата обращения: 09.08.2011).
4. Нильсен П. SQL Server 2005. Библия пользователя. – М.: Вильямс, 2008. – 1232 с.
5. Спирин Н.А., Лавров В.В., Бурыкин А.А., Краснобаев А.В., Быков А.Г. Технология и средства разработки информационно-моделирующих систем для решения технологических задач в металлургии // Известия Томского политехнического университета. – 2010. – Т. 316. – № 5. – С. 156–162.

Поступила 09.08.2011 г.