



# MES - теория и практика

Выпуск 2 (2010)

**Официальные материалы ассоциации  
MESA International**

WP 27. Сервис-ориентированная архитектура  
в системах управления производством

Как ошибки оценки в ERP помогают распространению MES

Стандарты и технологии интеграции производственных  
информационных систем

Глоссарий терминологии MESA



Издание официальное

Перевод выполнен  
русской рабочей группой  
MESA International  
[www.mesarussia.ru](http://www.mesarussia.ru)

Москва  
2010

УДК 658.51  
ББК 30.605

Выпуск одобрен Европейской штаб-квартирой ассоциации MESA International

Составители: И.С. Решетников, А.П. Козлецов

**MES - теория и практика.** Выпуск 2 (2010). Официальные материалы ассоциации MESA International. Москва, 2010.-80 с, илл.

Сборник содержит официальные переводы материалов ассоциации MESA International, в частности WP 27. Сервис-ориентированная архитектура в системах управления производством, Как ошибки оценки в ERP помогают распространению MES, Стандарты и технологии интеграции производственных информационных систем, Глоссарий терминологии MESA.

Сборник может быть полезен для руководителей промышленных предприятий, инженерно-технических работников, специалистов по автоматизации производственных процессов, студентов и учебных заведений.

УДК 658.51  
ББК 30.605

Документ переведен на русский язык при поддержке:



Группа компаний «СМС-Автоматизация»  
443035, г. Самара, пр. Кирова 201, секция 9  
Тел./факс: +7 (846) 933-03-50  
[www.sms-automation.ru](http://www.sms-automation.ru)



ООО «Компания «ТЕРСИС»  
адрес: Россия, г. Москва, 109082, ул. Солянка 1/2, стр. 1  
тел./факс: +7 (495) 980-73-57  
[www.tersys.ru](http://www.tersys.ru)



ООО «АМастер»  
адрес: Россия, г. Саратов, 410044, пр-кт Строителей, д. 1  
тел.: +7 (8452) 44-70-57  
тел./факс: +7 (8452) 44-70-70  
e-mail: [amaster@mail.saratov.ru](mailto:amaster@mail.saratov.ru)

© Российская рабочая группа MESA International, 2010

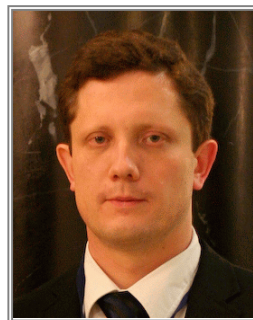
## СОДЕРЖАНИЕ

Предисловие .....	3
WP 27. Сервис-ориентированная архитектура в системах управления производством .....	5
Как ошибки оценки в ERP помогают распространению MES .....	61
Стандарты и технологии интеграции производственных информационных систем .....	68
Глоссарий терминологии MESA .....	85

## ПРЕДИСЛОВИЕ

ко второму выпуску издания «MES - теория и практика»

Когда готовился к выпуску первый выпуск издания, мы себе не очень представляли, что из представленной информации окажется востребованным. Вопреки нашим опасениям, брошюра была воспринята хорошо, и, судя по откликам, не раз помогала нашим читателям при организации процесса внедрения MES-систем, подготовке рекламных материалов (в т.ч. при внутреннем пиаре проекта) и др. задачах.



Читателями брошюры оказались очень разные люди – от руководителей высшего звена до студентов, причём не только технических, но даже гуманитарных специальностей. Некоторые хвалили, некоторые критиковали. Это не страшно, мы и не делали попытку выпустить Библию или бестселлер. Мы добились главного – издание оказалось полезным в реальной практической деятельности.

Формируя второй выпуск нашего издания, мы решили отойти от принципов построения первого, и собрать в нём уже не только переводы «Белых бумаг» или Whitepaper ассоциации MESA, но и несколько работ, которые, по нашему мнению, могут оказаться полезными для наших читателей.

Содержание второго выпуска подобрано так, чтобы дополнять первый выпуск, и составляет с ним единое целое. Имея оба выпуска, читатель получит информацию, которая поможет ему ориентироваться в идеологии и терминологии MESA, правильно подобрать методику внедрения и выбрать поставщика решений, оценить возможности современных технологий по интеграции на уровне систем и приложений, получить большое количество ценных рекомендаций.

Возможно, при беглом просмотре наши издания кажутся чем-то вроде «сборника очевидных советов» и вызывают даже некоторое недоумение. Но посмотрите на то, что происходит в реальных проектах. Куда девались эти знания? Почему многие, казалось бы, очевидные вещи не находят отражение в проектных решениях и организации проектов? Почему рекомендации «как надо» и «как не надо» оказываются перепутанными? Базовые знания элементарных основ при внедрении таких сложных и важных систем, как MES, важны и ими не стоит пренебрегать.

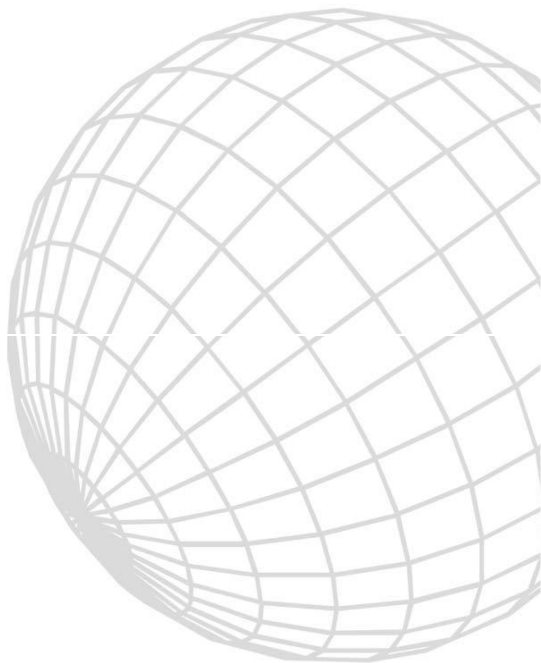
Именно поэтому мы предлагаем на Ваш суд второй выпуск издания «MES – теория и практика». По-прежнему мы с благодарностью хотели бы получить Ваши предложения и комментарии по улучшению этой серии, ваши пожелания по последующим выпускам сборника.

С уважением, И.С.Решетников,  
Руководитель Российской рабочей группы MESA International





# Сервис-ориентированная архитектура в системах управления производством



**WHITE PAPER 27**

## СОДЕРЖАНИЕ

Введение .....	7
Основные задачи и тенденции .....	8
Бережливое производство .....	8
«Шесть сигма» в производстве .....	9
Информационные технологии в производстве .....	9
Усиление эффекта при комбинировании подходов .....	10
Интересные инициативы для специалистов в области дискретного производства .....	12
Почему нужны гибкость, подвижность и способность воспринимать изменения .....	12
Системный взгляд на производственную компанию .....	14
Окружение .....	14
Производственная ИТ-инфраструктура .....	14
Беспроводные сети и взаимодействие .....	15
Клиентские и мобильные устройства .....	15
Интеграция устройств на производственном уровне .....	15
Многоуровневая модель компании .....	17
Производство 2.0 .....	19
Обзор сервис-ориентированной архитектуры .....	23
Что такое SOA? .....	23
Интеграция «точка к точке» .....	23
Интеграция корпоративных приложений .....	23
Сервис-ориентированная архитектура (SOA) .....	25
Компоненты сервис-ориентированной архитектуры .....	26
Сервисы .....	26
Сервисная шина .....	27
Координация процессов .....	27
Преобразование и маршрутизация сообщений .....	28
Реестр сервисов .....	28
Жизненный цикл SOA: моделирование, реализация, выполнение, управление .....	28
Моделирование .....	28
Реализация .....	31
Исполнение .....	32
Управление .....	33
Реестры и репозитории сервисов .....	33
Применение стандартов для достижения максимального эффекта от SOA .....	34
Стандарты на форматы и содержимое сообщений .....	36
Сервисные платформы .....	38
Дополнительные спецификации веб-сервисов .....	40
Заключение .....	43
Приложение А. Примеры успешного использования SOA .....	44
IBM Microelectronics Division .....	44
Система eHub компании Ford .....	50
Приложение В. Авторы и рецензенты руководства .....	54
Приложение С. Список сокращений/гlossарий .....	55
Приложение D. Товарные знаки .....	57
Приложение E. Список источников .....	58
Приложение F. Частичный список соответствия типов данных J2EE и .NET .....	59

## ВВЕДЕНИЕ

В связи с изменением бизнес-моделей производственные предприятия вынуждены быть более гибкими. Способность компании адаптироваться к изменяющимся условиям ведения бизнеса во многом зависит от гибкости корпоративной культуры и бизнес-процессов, а также от способности информационных систем, применяемых в компании, к взаимодействию. К сожалению, в настоящее время на многих предприятиях используются негибкие и устаревшие системы, которые трудно и дорого расширять, обслуживать и поддерживать.

Одна из причин изменения условий ведения бизнеса – рост числа нормативных документов, регулирующих деятельность производственных предприятий. В пищевой, аэрокосмической, оборонной, автомобильной промышленности необходимо иметь возможность проследить путь изделия «от колыбели до могилы», то есть от исходных материалов до конечного продукта. В некоторых случаях нужно прослеживать не только основную продукцию, но также и побочную продукцию, получаемую в ходе основных производственных процессов.

Другая причина, по которой гибкость предприятия становится всё более важной – участие в производстве конечного продукта большого числа поставщиков. Всё чаще OEM-производители используют готовые составные части, выпущенные участниками глобальной сети поставщиков, появляющимися так же часто, как и исчезающими. Для того, чтобы обеспечить рентабельность производства, компании должны иметь возможность легко и безопасно интегрировать свои информационные системы с информационными системами поставщика для отслеживания заказов, поставок, планов и т.д. Все эти информационные системы должны быть достаточно гибкими для того, чтобы поддерживать различные требования, регламентирующие деятельность разных OEM-производителей и поставщиков.

Одна из технологий (или архитектур) способных помочь компании в решении таких задач, носит название сервис-ориентированной архитектуры (Service-Oriented Architecture, SOA). Совместно с отраслевыми стандартами и методами непрерывного улучшения (continuous improvement, CI), использование SOA позволяет получить информационные системы, построенные по принципу “plug-and-play”. Отдельные функции в таких системах могут быть оперативно добавлены, изменены или удалены в соответствии с требованиями бизнеса.

В настоящем руководстве рассматриваются современные задачи и тенденции, возникающие в производственных отраслях, а также то, как эти задачи и тенденции вынуждают производственные предприятия переосмысливать ИТ-архитектуру. Обсуждается SOA и её составные части. Кроме того, рассказывается о новых инструментах, способных помочь компаниям в полной мере ощутить преимущества SOA. В зависимости от выбора технологии и платформы разработки, в понятие SOA может вкладываться несколько разный смысл. Наиболее популярные подходы состоят в применении Microsoft Windows Communication Foundation или Java 2 Enterprise Edition. Несмотря на то, что оба подхода схожи друг с другом, в них есть немало различий. В задачи настоящего руководства не входит сравнение двух технологий. С учётом опыта авторов, в руководстве предполагается использование J2EE и стандартов, использование которых общепринято в сообществе разработчиков Java.

## ОСНОВНЫЕ ЗАДАЧИ И ТЕНДЕНЦИИ

### *Бережливое производство*

Промышленное производство непрерывно развивается, и новые тенденции в организации производства постоянно появляются и исчезают. Они внезапно возникают в виде нового технического или социального вызова, на который производственные компании отвечают изменением корпоративной стратегии. Спустя некоторое время тенденция либо переходит в разряд общепринятых способов ведения бизнеса, либо замещается новыми инициативами и забывается. Однако есть одно исключение. Большинство предприятий, вынужденных из-за давления со стороны конкурентов использовать методы непрерывного улучшения, внедряют концепции бережливого производства (Lean Manufacturing). Особенно это касается автомобильной промышленности, так как само понятие бережливого производства впервые было сформулировано в рамках Производственной системы компании Toyota (Toyota Production System).

Концепции бережливого производства – важная часть бизнес-модели непрерывного улучшения. У крупных компаний, таких как General Motors или Ford, есть свои версии производственной системы Toyota. В других отраслях промышленности бережливое производство также активно используется. Основная задача бережливого производства состоит в сокращении потерь, что улучшает все ключевые показатели: качество, использование производственных мощностей, безопасность, использование материалов, стоимость, скорость поставки. В бережливом производстве любой сотрудник завода или служащий рассматривается как лицо, принимающее решение на вверенном ему участке работ. Каждый сотрудник учится определять узкие места производственных процессов и причины появления потерь, а также тому, как быстро устранять эти недостатки.

Бережливое производство – это не просто инструмент снижения стоимости или уровня складских запасов, как думают на многих предприятиях Америки. В бережливом производстве ставится задача создания гибкой структуры из организационных и технических систем, способной адаптировать рабочие процессы и цепочки поставок к мгновенным изменениям запросов рынка. Это значит, что компания должна иметь возможность обнаружить такое изменение и переконфигурировать ресурсы так, чтобы учесть наличие узких мест и возможности возникновения потерь, тем самым обеспечивая оптимальное выполнение работ и максимальную прибыль от каждого заказа. Преимущества, которых можно достичь в результате правильно реализованной программы бережливого производства хорошо описаны и документированы. Основные трудности связаны с необходимостью постоянно учитывать запросы рынка. В противном случае первоначальный эффект от реализации программы со временем теряется из-за отсутствия связи с реальным спросом. Другая проблема связана с необходимостью изменения подходов к менеджменту. Потребители ожидают неизменного уровня качества всей линейки продуктов. Невозможность обеспечения одинакового качества на разных заводах приводит к тому, что потребитель полностью отказывается от данного бренда и обращается к другому поставщику. Уже недостаточно иметь один завод мирового уровня – все заводы, входящие в компанию, должны быть такими.

Самые передовые производственные компании реализуют концепции бережливого производства на цеховом уровне и расширяют их на все остальные виды деятельности компании: складское хранение, логистику и т.д. Некоторые методы, такие как систематизация потока ценности (Value Stream Mapping), применяются в компании в целом. Передовые компании постоянно повышают производительность. Бережливое производство никогда не станет «обычным» способом ведения бизнеса. Улучшение процессов должно идти непрерывно и динамично в силу того, что его необходимость диктуется запросами рынка, изменением технологий, необходимостью выпуска новых видов продукции. Поэтому пространство для улучшения есть всегда.

## **«Шесть сигма» в производстве**

«Шесть сигма» - методология статистического анализа процесса с целью минимизации его вариаций. По сравнению с методами непрерывного улучшения, «шесть сигма» имеет более традиционный жизненный цикл, однако эту методологию очень легко использовать неправильно. Уроки, вынесенные из неудач с внедрением «шесть сигма» могут быть очень полезны при реализации концепций бережливого производства.

Одна из причин, по которым «шесть сигма» иногда внедряется сверху вниз состоит в необходимости развития культуры принятия решений на основе данных. На тренингах это буквально «вдалбливается» участникам. Естественная же тенденция состоит в том, чтобы принимать решения, базируясь на эмоциях и политических соображениях. Иногда решения принимаются на основе опыта, но многолетний опыт может значить и то, что полученные когда-то плохие привычки глубоко укоренились, и иногда этому способствуют даже некоторые нормативные документы.

Компания 3М успешно внедрила «шесть сигма» с помощью:

- общего языка, понятный всем участникам;
- принятия решений на основе данных;
- устранения барьеров между подразделениями.

СЕО компании обеспечил внедрение, сделав применение этих принципов безальтернативным, говоря всем, например: «Мы принимаем решения, основываясь на данных». Другие же говорят о том, что «шесть сигма» уничтожила все способности 3М к инновациям. Почему?

Развитие культуры принятия решений на основе данных – это изменение ключевых основ компании. Но результат может быть нивелирован в том случае, если слишком много внимания уделять инструментам и бездумно следовать всем шагам процесса внедрения. Вы должны знать, когда необходимо применять инструменты статистического анализа. Если основные факты и причины известны и без прохождения всех шагов «шесть сигма», без использования статистических инструментов, без следования всем бюрократическим процедурам и без недель и месяцев работы, то нужно просто действовать, вооружившись знанием этих фактов. Эффект от «шесть сигма» снижается и тогда, когда слишком много внимания уделяют финансовым и торговым процессам, забывая о разработке, снабжении, производственных процессах, процессах поставки и вспомогательных процессах. Очень часто разные процессы конфликтуют друг с другом и необходимо найти баланс между ними. Именно поэтому в Тойота для реализации любого проекта по непрерывному улучшению формируются команды из сотрудников нескольких отделов.

Вот недавняя запись из блога на [www.iSixSigma.com](http://www.iSixSigma.com): «Методология «Шесть сигма» была неправильно использована молодчиками с бумажными формами и гениями, которые не могли сделать ни шагу в сторону от стандартных процедур».

Независимо от того, выполнялось ли внедрение сверху вниз во всей компании, или только в отдельных подразделениях, «шесть сигма» и принятие решений на основе данных предъявляли дополнительные требования к надёжности сбора и корректности данных. Также требовалось специальное программное обеспечение для статистического анализа или других методов «шесть сигма» и бережливого производства.

## **Информационные технологии в производстве**

В последние десять-пятнадцать лет всё сильнее проявляется тенденция к широкому и всеобъемлющему распространению компьютерных технологий на производстве. В результате снижения стоимости покупки и развёртывания мощных серверов и программных пакетов производственные предприятия решают с их помощью гораздо большее число задач, чем в начале 90-х годов. На некоторых заводах сейчас применяются сотни различных приложений, часть из которых выполняется на процессорах, встроенных в

производственное оборудование, в то время как другая часть функционирует в центрах обработки данных и используется руководством предприятия.

На первых этапах на производстве использовалось множество отдельных программных пакетов, различавшихся моделью данных, архитектурой приложения, используемыми транзакциями и структурой сообщений. Стоимость владения такими приложениями очень высока вследствие большого разнообразия, недостаточной гибкости и трудности интеграции. В результате на многих предприятиях в дополнение к уже имеющимся программным средствам для сбора и анализа данных используются электронные таблицы и текстовые документы. Как следствие невозможно за разумные средства реализовать хранение, сопоставление, анализ данных, а также применение результатов всех этих действий. По данным AMR Research, потребность в интегрированной ИТ-архитектуре можно проиллюстрировать следующими цифрами: доля средств в ИТ-бюджете предприятия, необходимых для поддержки производственных операций, возросла с 3% в 2001 году до 19% в 2007 году.

Стратегия использования ИТ в производстве должна формироваться в соответствии со бизнес-стратегией непрерывного улучшения. Отдел ИТ и производственный отдел должны реализовывать корпоративную стратегию в своих производственных стратегиях и системах. На самых успешных производственных предприятиях отделы ИТ уделяют основное внимание поддержке улучшений рабочих процессов. В то же время видно, что большее внимание при разработке ИТ-стратегий и архитектур следует уделять определению и улучшению производственных метрик, способных помочь в представлении текущего состояния производства. В автомобильной отрасли, например, такими метриками могут быть время запуска в производство новой трансмиссии или автомобиля, первоначальное качество, снижение отходов и числа гарантийных случаев, использование производственных мощностей и время доставки продукции.

Основная задача ИТ при поддержке бережливого производства – обеспечение лиц, принимающих решения (напомним, что это все сотрудники предприятия) корректной информацией, достаточной для принятия верных и своевременных решений. На некоторых заводах, выпускающих большие объёмы продукции и обладающих высокой степенью автоматизации уже используются устройства, способные в реальном времени определять, анализировать и исправлять проблемы с производственными процессами. Логика выполнения бизнес-процессов постепенно начинает встраиваться в информационные системы. Концепция SOA ускоряет внедрение «бережливых» концепций в ИТ. Для того, чтобы сделать улучшения в рабочих процессах постоянными, необходимы изменения в ИТ-системах. Со временем позитивные изменения накапливаются и приводят к получению значительных серьёзных преимуществ в бизнесе.

### ***Усиление эффекта при комбинировании подходов***

В докладах руководителей производственных предприятий на конференциях, проводимых ARC Advisory Group, отмечалось усиление эффекта при комбинировании методов непрерывного улучшения с расширением функциональности информационных систем, методами бережливого производства, развитием связей и взаимодействия с персоналом, применением отраслевых стандартов, методами «шесть сигма» и лучшей информированностью. В таблице 1 приведены примеры комбинирования методов непрерывного улучшения и использования информационных систем для того, чтобы дать производству возможность более гибко адаптироваться к глобальному рынку.

Таблица 1. Примеры непрерывного улучшения, усиленного информационными технологиями

Подход	Результаты
<p>Выполнена переработка производственных процессов, в результате чего обнаружено, что ИТ-архитектура фрагментирована. Для решения проблемы выбрана единая платформа – программный пакет ERP. Факторы улучшения производительности: автоматизация, лучшее использование человеческого капитала, лучшая организация управления, лучшие ИТ-инструменты.</p> <p>«Это изменение способа управления с самого верха: модель лидерства, в которой вы полностью полагаетесь на людей рядом с вами. Вы должны взаимодействовать с людьми, должны увлекать их и у вас должны быть цели и метрики, с помощью которых вы сможете судить о близости целей. Эти понятия не являются специфичными для ИТ, но ИТ жизненно важны для многих из них. Перед тем, как даже задуматься об ИТ-архитектуре, нам пришлось понять стратегические планы компании»</p>	<p>На 80% снижено время реакции для каждой транзакции, в то время как объем транзакции вырос на 450%. С 1996 года завершено более 2000 ИТ-проектов. Их объем с 2006 года составил более 2,5 миллиардов долларов. Фиксация уровня затрат на ИТ (1997-2004). За период 1992-2004 производительность росла на 8% в год.</p>
<p>Общее программное обеспечение (единое семейство ПЛК), общее аппаратное обеспечение (стандартные панели), стандартные открытые информационные сети, курсы обучения, протектирование систем и имитационные приложения, усиливающие общность. Архитектура систем управления, масштабирующаяся для заводов различных размеров.</p>	<p>Экономия от 50% до 70% затрат на оборудование систем управления. Снижение стоимости развертывания и обслуживания производственных приложений во всей компании.</p>
<p>Единая ИТ-стратегия, приведшая к необходимости использования MES для расширения возможности управления производственными процессами и определения участков для улучшения. Необходимо начинать именно с единой стратегии. Последующие преимущества возникают в результате создания культуры принятия решений на основе данных с особым вниманием на непрерывное улучшение. Очень трудно управлять чем-либо с плохими данными.</p>	<p>Снижение времени цикла с 4 недель с рентабельностью -4% в 2001 г. до 5 дней с рентабельностью 69% в 2006. 100% своевременная доставка на протяжении нескольких лет. Стоимость одного заказа сократилась на 25%.</p>
<p>В концепции бережливого производства требование устойчивости ограничивает скорость улучшений. MES жизненно важна для устойчивости процесса. С помощью MES вы можете наблюдать за данными о выходе годных изделий, данными о параметрах процесса, данными об обнаружении дефектов и их причин, а также управлять этими данными.</p>	<p>Концепции бережливого производства и «шесть сигма» полностью поддерживаются заводскими управленцами. MES обеспечила для них нужную степень видимости производственных процессов. Результаты на одном заводе: увеличение производительности на 60%, снижение уровня складских запасов на 50%. Результаты на втором заводе: увеличение производительности на 40%.</p>
<p>Бережливое производство: протектирование технологической линии, материальные и информационные потоки, изменение в системе поощрений и содержании работ. Не ограничивайтесь следованию первоначальным целям протектирования информационных систем и не используйте функции программного обеспечения только потому, что они есть. Используйте только то, что помогает бизнесу двигаться вперед.</p>	<p>Высокоэффективное внедрение недавно приобретённой ERP-системы в процессе внедрения бережливого производства. Снижение числа ошибок поставки на 70%. Все сотрудники, связанный с поставками, заканчивают рабочий день в 5 часов вечера. Время производственного цикла сокращено с 1,5 недель до 4 часов. Значительный рост продуктивности работы позволил компании выйти на рынки, ранее закрытые для неё.</p>
<p>Отслеживание материалов на всех этапах производственного процесса. Соответствие требованиям нормативных актов, обнаружение потерь продукции, измерение использования активов, отслеживание количества и стоимости материалов и инструментов, отслеживание отходов, определение выхода годных изделий. Безопасные и точные данные для бухгалтерского учёта.</p>	<p>Увеличение рентабельности и соблюдение требований законодательства. Снижение уровня потерь материалов с 2,5% до 0,75%. Снижение уровня складских запасов. Возможность обнаружить проблему ещё до того, как она выйдет за пределы производственного участка. Устранение ненужного расхода средств вследствие более эффективного использования оборудования.</p>

## ***Интересные инициативы для специалистов в области дискретного производства***

Такие тенденции, как использование концепций бережливого производства рождаются потому, что какая-то компания умеет что-то делать гораздо лучше, чем конкуренты. Постепенно её опыт перенимают и остальные участники рынка. Другие тенденции появляются в результате развития социальных процессов или развития технологий за пределом мира производственных предприятий. В качестве примера можно привести глобализацию. Повышение функциональности информационных систем, а также развитие экономики и инфраструктуры в таких странах, как Китай, Индия и Бразилия привели к появлению беспрецедентной возможности для компаний использовать материалы и сервисы из любой точки мира, производить продукцию где угодно и продавать её в любой стране.

Новые тенденции появляются также и в результате совместной работы ведущих производственных компаний, правительственных агентств и университетов. Ниже приведены примеры таких инициатив для дискретного производства.

1. Семинар по «умным» технологиям сборки в октябре 2006. Семинар проведён национальным институтом стандартизации (NIST). В семинаре участвовали представители Ford, GE, Boeing, GM и других компаний. Контактное лицо – Дейл Холл (Dale Hall), руководитель лаборатории проектирования в производстве, NIST.
2. Федеральная межведомственная рабочая группа по НИОКР в производстве определила направление «Интеллектуальные и интегрированные производственные системы» как одно из ключевых направлений национальных интересов.
3. Совместный исследовательский центр по интеллектуальным производственным системам при Национальном научном фонде (NSF). Среди участников – Ford и Toyota. Основная цель работы – устранение простоев. Контактные лица: Джей Ли (Jay Lee), выдающийся учёный Огайо<sup>1</sup>, профессор кафедры перспективных технологий производства, Университет Цинцинатти, Jay.Lee@uc.edu. В работе Центра также принимают участие Университет Мичигана и Университет Миссури-Ролла.
4. Группа по развитию автомобильной отрасли (The Automotive Industry Action Group, <http://www.aiag.org>, отделения в Саутфилде, США и Шанхае, Китай) также имеют в своём составе группы сотрудников автомобильных компаний, правительственных агентств и других организаций, работающих совместно над следующими инициативами:

А. Прозрачность и взаимодействие на складах (Inventory Visibility & Interoperability, IV&I):

- фаза 1: MIN/MAX, профиль для веб-сервисов;
- фаза 2: eKanban и надёжный и безопасный обмен сообщениями;
- фаза 3: склад, управляемый поставщиком и надёжный и безопасный обмен сообщениями.

В. Стандарты раннего предупреждения.

С. Взаимодействия производства и бизнеса.

Д. Работа с зарубежными поставщиками материалов.

## ***Почему нужны гибкость, подвижность и способность воспринимать изменения***

За последние 15 лет стандарты обмена информацией между производственными и корпоративными информационными системами прошли путь от высокоуровневых моделей бизнес-процессов до чётко специфицированных схем, определяющих последовательность передачи и структуру сообщений. Однако внедрение таких стандартов происходит довольно медленно и тому есть две причины. Во-первых, ощущается недостаток примеров успешного

---

<sup>1</sup> Ohio Eminent Scholar – почётное звание исследователей, работающих в штате Огайо



использования стандартов для вертикальной интеграции. Во-вторых, стоимость замены устаревших систем, основанных на различных моделях данных и интерфейсах «точка к точке» остаётся высокой.

Наиболее прогрессивные компании добились существенных успехов, определив стандарты для компьютерных приложений и их интеграции с использованием канонических схем и наборов транзакций. Также как при внедрении концепций непрерывного улучшения, для таких изменений в ИТ-архитектуре требуется 3-5 лет для приведения приложений и интерфейсов между ними в соответствие с установленным стандартом. Процесс миграции значительно ускоряется с применением технологий SOA. Они помогают снизить как величину первоначальных инвестиций, так и стоимость обслуживания различных систем.

На производственном предприятии 21-го необходим двусторонний обмен между корпоративным и производственным уровнем. В настоящее время интенсивно развиваются методы агрегации и анализа данных, которые должны обеспечить авторизованный доступ только к той информации, которая действительно необходима. Для отображения агрегированных данных используются инструментальные панели, специфичные для данного пользователя или роли, порталы и переносные персональные устройства. Отображаемая информация должна соответствовать потребностям, задачам и уровню доступа каждого пользователя.

Цепочки поставок, объединяющие предприятия по всему миру, требуют возможности открытой и безопасной связи с любым поставщиком. В ситуации, когда многие производственные операции сопровождаются «бумажными» транзакциями, а компьютерные приложения в цеху и офисе изолированы друг от друга, одной из главных задач становится унификация и интеграция фрагментированных и чрезмерно сложных ИТ-систем в каждом регионе, в каждой компании, на каждом заводе и производственной линии. Для того, чтобы достичь нужного уровня адаптируемости, необходимо весь набор систем, применяемых компанией и её поставщиками, встроить в единую архитектуру на основе SOA.

В проекте по взаимодействию производственного и корпоративного уровней (P2B) Группы развития автомобильной промышленности, на основе анализа опыта различных компаний определено следующее.

«Общепринятые стандарты по взаимодействию между производственными и корпоративными системами, а также системами управления цепочками поставок в настоящее время отсутствуют. В результате необходимы большой объём первоначальных инвестиций, а стоимость обслуживания различных систем становится очень высокой. Возможность взаимодействия с производственным уровнем становится ещё более критичной вследствие следующих причин:

- глобализация требует прозрачного взаимодействия в реальном времени со всеми участниками цепочки поставок;
- производственные системы становятся всё более сложными, а номенклатура выпускаемой продукции – всё более разнообразной;
- в качестве поставщиков выступают как внешние предприятия, так и подразделения внутри компании, что требует большей степени интеграции данных и безопасности связи;
- необходимо обеспечивать обмен информацией между производственным и корпоративным уровнем, при этом нужно, чтобы необходимая информация была доступна только авторизованным пользователям; для этого необходимы настраиваемые инструментальные панели;
- открытые и безопасные коммуникации между участниками цепочки поставок во всём мире – важнейшая часть системы. Объединённые производства комплектующих, интеграция поставщиков в производственные мощности OEM, развёртывание OEM-производителями своих систем на предприятиях поставщиков – всё это требует затрат как со стороны OEM-производителя, так и со стороны поставщиков;

- на разных производственных линиях, заводах, в разных компаниях и регионах используются фрагментированные, чрезмерно сложные ИТ-архитектуры и элементы инфраструктуры информационных систем;
- необходимо снизить стоимость, сократить время и увеличить качество в производственных и корпоративных информационных системах;
- стоимость отзыва продукции и обслуживания гарантийных случаев должна держаться под контролем путём сосредоточения на качестве продукта и процесса, прослеживаемости компонентов, оптимизации управления складом по всей цепочке поставок;
- оптимальное использование материалов для минимизации складских запасов, что является ключевым в реализации концепций бережливого производства и соответствующего снижения затрат. Для того, чтобы нужная деталь оказалась в нужное время в нужном месте особо критичными становятся информационные потоки и интеграция;
- стоимость обслуживания и ремонта множества разрозненных систем становится очень высокой;
- отсутствуют стандартные способы связи с различными ПЛК несмотря на важность такой связи;
- оборудование радиочастотной идентификации (RFID) и беспроводные устройства ещё больше повышают сложность интеграции;
- при внедрении новой системы необходимо в сжатые сроки обеспечить её интеграцию с существующими системами.

Взаимодействие между корпоративным и производственным уровнями – ключ к обеспечению синхронизации и видимости процессов. Высшее руководство многих компаний, в том числе Microsoft, Oracle, IBM и SAP, всецело поддерживает стандарты. Однако стандарты требуют дальнейшей проработки, необходимо также развивать методологии обеспечения взаимодействия между различными системами.

Движение к использованию бизнес-стандартов – процесс небыстрый, но крайне важный. Для эффективного анализа, принятия решений и планирования развития стандартов необходимо знать мнение конечных пользователей. Слабая поддержка среди поставщиков seriously сдерживает распространение таких стандартов.

## **СИСТЕМНЫЙ ВЗГЛЯД НА ПРОИЗВОДСТВЕННУЮ КОМПАНИЮ**

### ***Окружение***

#### **Производственная ИТ-инфраструктура**

На производственном предприятии можно выделить три зоны, в которых используются информационные технологии:

- офисы;
- производственные помещения;
- центр обработки данных предприятия (серверная комната или вычислительный центр).

Такая ситуация типична. Офисные работники выполняют свои функции с использованием программного обеспечения и инфраструктуры, одинаковых на всех заводах компании. Они используют стандартные клиентские рабочие места на базе ПК, подключенных к офисной информационной сети. Рабочие места располагаются, в основном, в офисных помещениях предприятия, а также в помещениях для совещаний (рядом с производственными помещениями). Офисные информационные сети обычно доступны и в производственных помещениях.

Программируемое оборудование, ПЛК, координатные машины, оборудование для подачи материалов, испытательные стенды и другое оборудование расположено, в основном, в производственных помещениях. Мобильные устройства могут использоваться в любом месте на территории завода. Всё это оборудование подключается к специальным информационным сетям, отделённым от офисных информационных сетей для обеспечения безопасности персонала и данных.

На каждом предприятии имеется серверная комната с фальшполом, несколькими независимыми вводами электроснабжения, источниками бесперебойного питания, системой кондиционирования и т.д., используемая как небольшой центр обработки данных. Серверы и другие устройства доступны клиентам посредством офисной информационной сети, хотя могут использоваться и клиентами, подключёнными к промышленным сетям. К некоторым серверам доступ из офисной информационной сети может быть ограничен специально.

### **Беспроводные сети и взаимодействие**

На производстве используется большой набор технологий: радиочастотная идентификация (RFID), метки, датчики и т.д. Как беспроводные, так и традиционные медные или оптические сети предоставляют возможность взаимодействия путём передачи данных, голоса, мультимедийной информации как внутри производственных помещений, так и между производством и офисами. Протоколы передачи голосовых данных через интернет (VoIP), видеоконференции, службы мгновенных сообщений также могут быть полезны, но их применение зачастую ограничивается по соображениям безопасности. Инфраструктура для реализации защищённого взаимодействия между сотрудниками на всей территории предприятия и смежных территориях, инженерами управленческого аппарата, командами проектировщиков продукции и поставщиками материалов всё ещё разрабатывается.

### **Клиентские и мобильные устройства**

Стандартные офисные информационные технологии должны использоваться везде, где только возможно. Однако производственным предприятиям могут быть нужны офисные принтеры, разнообразные специализированные плоттеры, принтеры этикеток и штрих-кодов. Для использования на производстве могут потребоваться различные варианты исполнения устройства. Например, зачастую требуется, чтобы устройство имело специализированное промышленное исполнение, делающее его устойчивым к влиянию вредных внешних факторов. С другой стороны, в большинстве случаев для управления устройством используются стандартные технологии и операционные системы (Windows CE, Embedded Java и т.д.).

Обычно используется:

- аппаратное обеспечение киосков и терминалов;
- клиентские устройства для управления оборудованием (без пользовательского интерфейса);
- клиентские устройства с пользовательским интерфейсом (для управления испытательными стендами и т.д.);
- производственное оборудование со встроенной системой управления на основе ПК.

### **Интеграция устройств на производственном уровне**

Для преобразования закрытых форматов представления данных, используемых в существующем оборудовании и системах управления технологическим процессом, в стандартные сообщения могут использоваться OPC-сервера (OPC – OLE for Process Control). Со временем для связи с оборудованием можно будет использовать веб-сервисы, определённые организацией WS-I (Web-Service Interoperability). В этом случае будут доступны для применения новейшие спецификации безопасного и надёжного обмена сообщениями. С учётом таких тенденций организацией OPC Foundation разработана новая

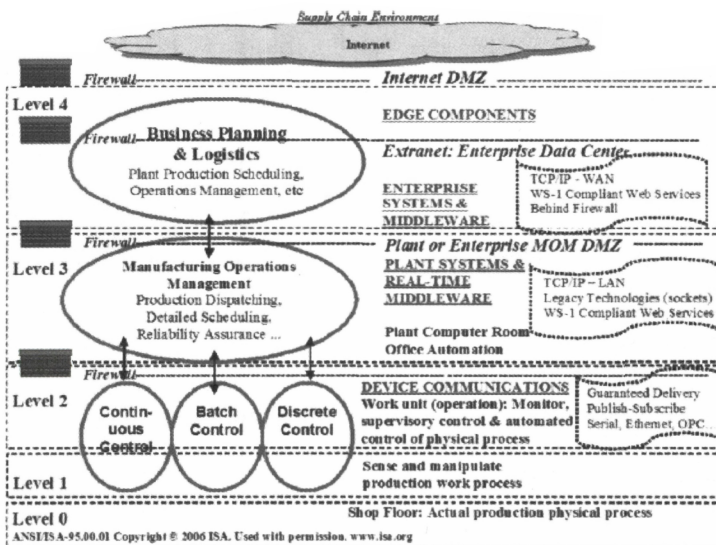
спецификация под названием «OPC Unified Architecture» (OPC UA). В OPC UA допускается несколько различных вариантов реализации на основе J2EE, Microsoft .NET и OPC UA Binary. Применение веб-сервисов и OPC позволяет полностью пересмотреть ИТ-инфраструктуру, добиться принципиально нового уровня общности и стандартизации и в то же время обеспечить поддержку производственных систем, возраст которых зачастую достигает 10-15 лет.

На рис. 1 показано логическое представление производственного предприятия в цепочке поставок. Предприятие делится на несколько уровней в соответствии с моделью ISA-95: производство, центр обработки производственных данных, технологическая сеть предприятия или компании, корпоративный центр обработки данных, экстранет, интернет. Для логического разделения уровней используются коммуникационные шины или сетевые экраны. При этом коммуникационные шины объединяют все уровни между собой, а между некоторыми уровнями располагаются сетевые экраны. Для связи могут использоваться одна или несколько различных коммуникационных шин. На рисунке деление производится по типу коммуникаций, применяемых на каждом уровне. Например, для связи между ПЛК и другими устройствами на производственном уровне может использоваться как OPC, так и другие технологии (сокеты и т.д.).

На уровне 4 отслеживается и контролируется уровень запасов, а также выполняется первоначальное планирование работы предприятия – планирование производства, использования материалов, приёмки и отгрузки. Временные промежутки, с которыми работают на данном уровне, составляют дни, недели и месяцы.

На уровне 3 определяются ресурсы и оборудование, необходимые для производства продукта и выполняется управление производственными процессами и рецептами. Системы управления производственными операциями (Manufacturing Operations Management, MOM), в том числе и MES анализируют данные о ходе производства, управляют сохранением данных и оптимизируют производственные процессы. На этом уровне оперируют такими временными промежутками, как дни, смены, часы, минуты и секунды.

**Рис. 1.**  
**Иерархия**  
**ISA-95**  
**(модель**  
**университета**  
**Пердью)**



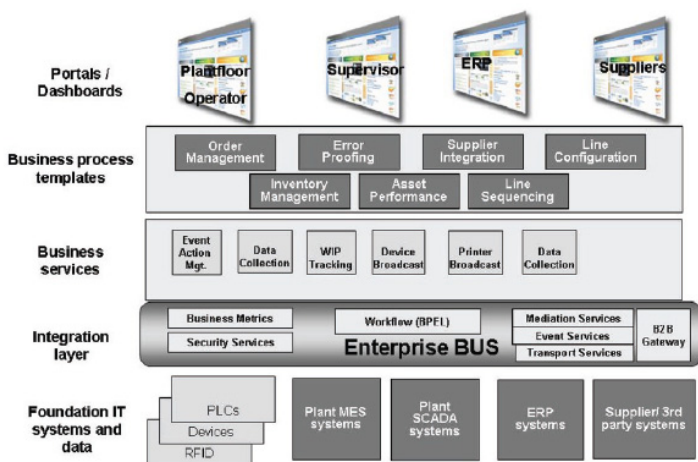
На уровне 2 определяются операции, необходимые для производства продукта. Основная задача систем, работающих на данном уровне – мониторинг производства и реализация диспетчерского или автоматизированного управления производственным процессом. Временные промежутки – часы, минуты, секунды, микросекунды и ниже.

На уровне 1 производятся измерения и формируются управляющие воздействия, необходимые для управления производственным процессом. Устройства на уровне 1 работают в реальном времени, типичные временные промежутки составляют микросекунды и ниже.

К уровню 0 относятся непосредственно производственные процессы, все системы управления находятся на более высоких уровнях.

Во многих компаниях необходимо, чтобы завод работал 24 часа в сутки семь дней в неделю даже при отсутствии соединения с Интернет и другими информационными сетями. В таких условиях необходимо размещать системы, работоспособность которых критична для производства, непосредственно на заводе. Такой подход типичен для ресурсоёмких массовых производств с непрерывными производственными процессами.

**Рис. 2. Обзор SOA для производства**



## Многоуровневая модель компании

Сервис-ориентированная архитектура зачастую может быть представлена в виде многоуровневой модели, приведённой на рис. 2. Нижний уровень состоит из унаследованных или существующих приложений, формирующих основу для обработки данных о работе компании. Здесь можно встретить важнейшие приложения, обеспечивающие непрерывную работу компании. Следующий уровень обеспечивает интеграцию. В SOA уровень интеграции представлен сервисной шиной предприятия (Enterprise Service Bus, ESB). Уровень ESB предоставляет сервисы для передачи информации, её преобразования, обеспечения безопасности передачи и обработки сообщений. Здесь же может быть реализовано вычисление бизнес-метрик, а также иметься средства для моделирования потоков работ или бизнес-процессов. Уровень бизнес-сервисов предназначен для абстрагирования от приложений первого уровня. На этом уровне находятся сервисы, играющие роль «обёртки» для таких приложений. Они описываются с помощью языка WSDL (Web Service Description Language, WSDL). Следующий уровень состоит из бизнес-процессов, которые создаются путём соединения отдельных сервисов уровня бизнес-сервисов для создания сложных составных приложений. Составные приложения – это новый

способ разработки приложений с использованием концепций SOA (более подробно он описан в разделе «Координация процессов» главы «Обзор SOA»). Уровень порталов и инструментальных панелей состоит из средств агрегирования и визуализации данных.

При использовании концепции SOA о сервисах можно сказать следующее:

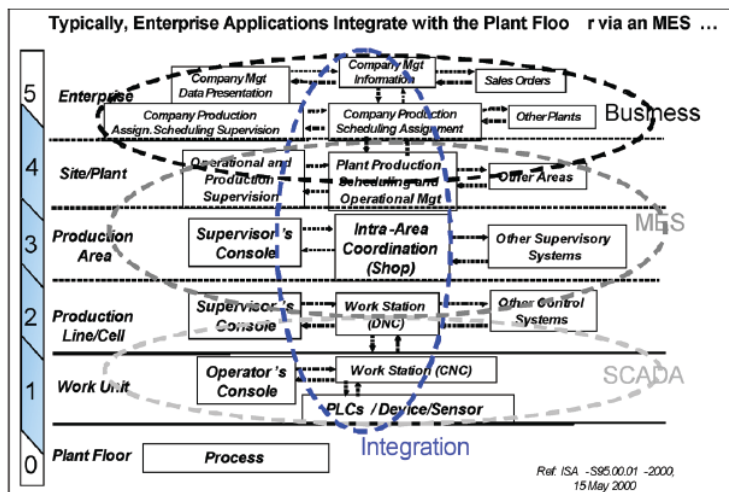
- они решают бизнес-задачи самостоятельно или с использованием координации процессов;
- используются внутри портала или веб-приложения для агрегирования или визуализации данных;
- новые визуальные приложения могут запускать один или несколько сервисов на выполнение (с использованием составных приложений).

Всю компанию можно разделить на уровни, управляемые приложениями одного из трех типов:

- бизнес-приложения или корпоративные приложения, такие как модули ERP-системы для финансового учёта и формирования отчётов, управления заказами, управления закупкой материалов и т.д.;
- MOM-системы или MES, координирующие производство и связывающие корпоративные и производственные приложения;
- приложения уровня оборудования, такие как распределённые системы управления (DCS) или автоматизированные системы диспетчерского управления и обработки данных (SCADA), которые применяются для управления производственным оборудованием.

Схема на рис. 3 построена в соответствии с моделью ISA-95 и показывает типичные места этих трёх классов приложений.

Рис.3. Типичная системная структура производственного предприятия



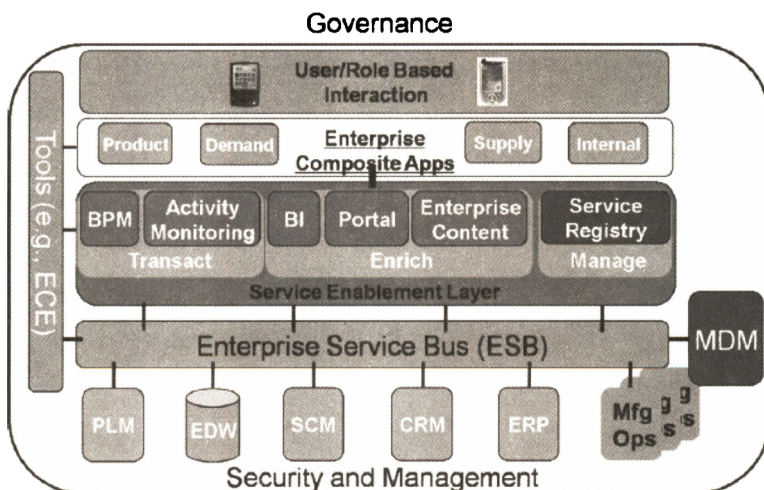
В настоящее время большинство производственных предприятий можно описать с использованием модели ISA-95. Однако многие компании переосмысливают способы интеграции данных и потоки данных между уровнями модели. SOA может сыграть огромную роль в обеспечении такой интеграции.

## Производство 2.0

Компанией AMR Research разработан подход к внедрению концепций SOA с учётом специфических требований производственных предприятий. Между реализацией SOA с использованием ESB для компании в целом и применении SOA для работы в масштабе времени, близком к реальному, с использованием сервисной шины производства (Manufacturing Service Bus, MSB), имеются серьёзные различия. Системы управления производственными операциями с использованием SOA получили общее название «Производство 2.0» для того, чтобы подчеркнуть их отличие от «Производства 1.0», при котором основу ИТ-архитектуры составляли обособленные клиент-серверные приложения, с помощью которых пытались моделировать бизнес-процессы и производственные процессы. Для взаимодействия в таких приложениях применялись двухточечные (point-to-point) интерфейсы, а преобразование данных выполнялось в зависимости от того, какие приложения поставляли или использовали эти данные. Цель данного раздела – краткое введение в «Производство 2.0» и другие элементы SOA. Более подробно SOA рассмотрена в последующих разделах руководства. Производство 2.0 рассматривается для того, чтобы у читателя сложилось общее представление о том, как SOA может использоваться как в компании в целом, так и на производственных участках.

На рис. 4 приведено содержание рис. 2 с точки зрения AMR. Под системами управления производственными операциями (Mfg Ops) понимаются MES и SCADA. Рис. 4 – основа для рисунков 5-7. Системы управления производственными операциями подробно описаны на рис. 5. Здесь приведены основные элементы и связи, входящие в концепцию «Производство 2.0», и обеспечивающие выполнение производственных операций затрагивающих один или несколько производственных участков.

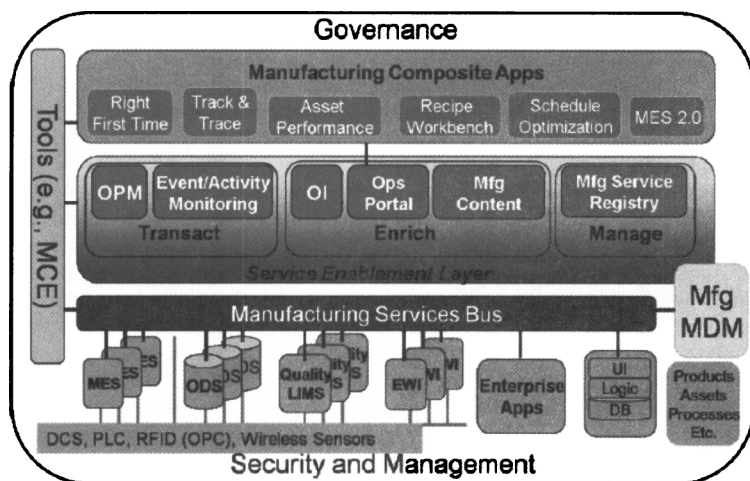
Рис.4.  
Корпоративная SOA



В руководстве «Производство 2.0» рассматривается как один из возможных эволюционных путей внедрения SOA для поддержки производственных операций, близких к реальному времени. Для более глубокого знакомства Вам следует обратиться к оригинальным работам AMR. Из рис. 5 видно, что для транзакций высокой интенсивности, передачи больших объемов данных и реализации прочих требований приложений, работающих в реальном времени, необходима специальная сервисная шина – MSB. Область



**Рис.5.**  
**Производственная SOA**



использования MSB может быть сокращена до одного завода или цеха в зависимости от требований к скорости и объёму данных, передаваемых в ходе производственных процессов. При рассмотрении «Производства 2.0» важно понимать, что управление основными производственными данными (Mfg MDM) отличается от управления данными на уровне компании. Сервисы управления нормативно-справочной информацией – это набор приложений для управления такими производственными операциями, как диспетчеризация, управление перемещением продукции и материалов, обработка аварийных сообщений. Объекты, с которыми работают эти приложения, имеют гораздо большее количество атрибутов, чем те объекты, с которыми работают приложения для обработки основных данных на корпоративном уровне (общее планирование, логистика). Однако управление основными данными – слишком сложная и объёмная тема для того, чтобы быть адекватно раскрытой в настоящем руководстве. Здесь мы ограничимся кратким описанием, необходимом для определения одного из важнейших компонент SOA. Для более серьёзного знакомства советуем обратиться к книге «Enterprise Master Data Management: An SOA Approach to Managing Core Information», <http://safari.oreilly.com/9780137149674>.

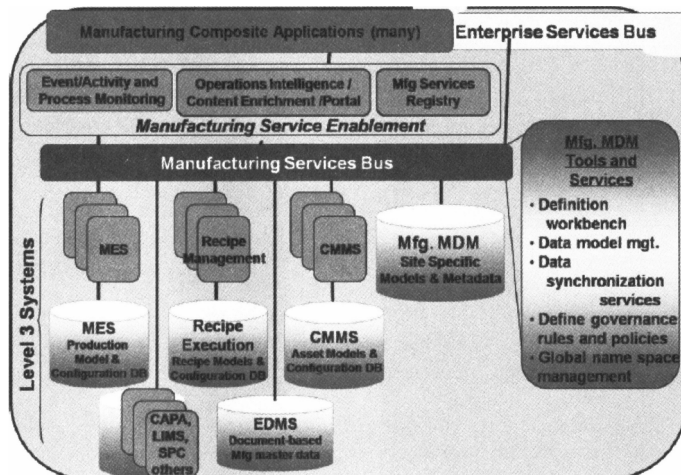
Способ организации и роль управления основными данными сильно зависят от отрасли промышленности, номенклатуры продукции, сегмента рынка, типа и сложности производственного процесса, а также от типа цепочки поставок. Например, управление основными данными во многом отличается в компаниях, занимающихся производством биопродуктов, компаниях, работающих в автомобильной или авиакосмической отраслях, компаниях электронной промышленности и т.д.

На рис. 6 показано, что эти различия сконцентрированы в специфических моделях и метаданных, применяемых на конкретном предприятии. Из-за высокой скорости изменения приложений «Производства 2.0» при выпуске новой продукции, изменении процедур учёта запасов, изменении документов, сопровождающих производство, и масштабировании производства в состав средств управления основными данными следует включить следующие инструменты и сервисы:

- определение продукции;
- управление моделями данных;
- сервисы синхронизации данных;
- определение правил и политик управления;
- управление общим пространством имён.



**Рис.6. Управление производственной нормативно-справочной информацией в SOA**



На рис.7 показано место SOA на производстве в общей концепции SOA на предприятии. «Производство 2.0» - это подход на основе SOA, а не компьютерное приложение. Подход учитывает разнообразие, сложность, и высокую динамику в «запросо-ориентированном» производстве и при этом:

- устраняется недостаток производственного опыта;
- сохраняется и даже увеличивается объём инвестиций в производство, что позволяет избежать постоянной замены одних недостатков другими;
- устраняются особенности пользовательского интерфейса и жёсткие модели данных, мешающие созданию, развёртыванию и модификации производственных приложений;
- расширяется совместная работа над продуктами и производственными процессами как внутри предприятия, так и совместно с партнёрами.

При таком подходе приложения могут легко и недорого адаптироваться к изменению бизнес-процессов. Ниже перечислены самые впечатляющие характеристики «Производства 2.0».

#### 1. Интерфейсы, ориентированные на пользователя:

- задачи решаются наиболее простым способом;
- простая навигация и доступ к функциональности одних приложений из других посредством одинаковых интерфейсов (например, управление качеством и MES, эффективности использования оборудования (OEE) и управление производственными активами, системы предупреждающего корректирующего действия (CAPA) и системы управления лабораторной информацией (LIMS), LIMS и MES);
- возможность использования знаний, имеющихся у специалистов производства, при развёртывании, переконфигурировании и сопровождении программного обеспечения;
- стандартный пользовательский интерфейс в соответствии с рекомендациями Microsoft, возможность получения справки в ходе выполнения конкретных процедур;
- применение хорошо себя зарекомендовавших способов организации взаимодействия и адекватных информационных моделей;
- графические модели оборудования, возможность использования GPS и навигации внутри завода, широкие возможности представления данных, применение мультимедиа.

Приложения функционируют в соответствии с представлением пользователя о своей работе: функциональность приложения «отображается» в контексте бизнес-процессов или производственных процессов.

2. Производственные архитектуры, которые:

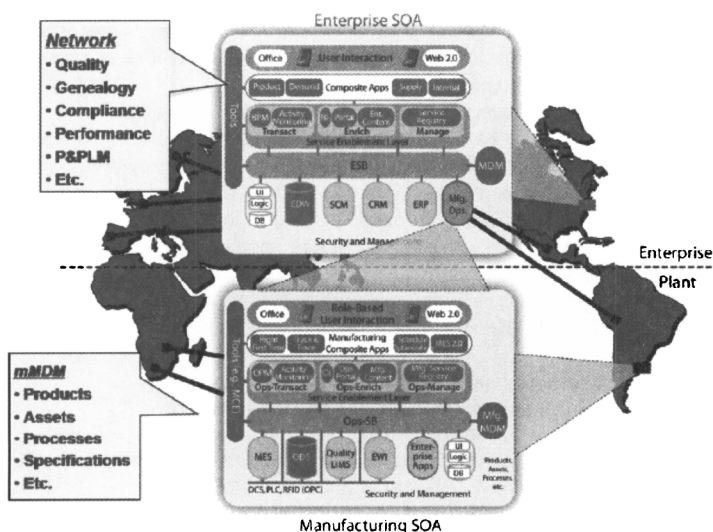
- усиливают эффект от инвестиций в существующие приложения, так как эти приложения встраиваются в SOA, а не заменяются на новые монолитные;
- используют преимущества моделей ISA-95/OAGIS;
- широко применяют возможности сервисной шины производства.

3. Соединение веб-технологий и корпоративных технологий: блоги, мгновенные сообщения, комбинированные приложения, поиск, вики-технологии, постоянно подключенные мобильные устройства;

4. Привлечение нового поколения, не мыслящего себя без интернет, в производственную сферу; повышение удобства использования программного обеспечения, расширение возможностей совместной работы и обмена знаниями;

5. Поддержка дешёвых идентификационных меток, интеллектуальных датчиков, всепроникающих информационных сетей, повышение мобильности сотрудников. Мобильные пользователи напрямую взаимодействуют с производственной средой через приложения для управления активами и качеством, MES и другие составные приложения, компоненты которых развёрнуты на мобильных устройствах.

**Рис.7. Производственная SOA как часть корпоративной SOA**



6. Платформа для взаимодействия с участниками цепочки поставок. Интернет предоставляет платформу с беспрецедентными возможностями по организации взаимодействия между поставщиками и производителями, производителями и конечными заказчиками.

7. Совмещение средств управления данными о ходе производства и средств разработки продуктов и процессов, что даёт возможность сократить время выхода на рынок.

# ОБЗОР СЕРВИС-ОРИЕНТИРОВАННОЙ АРХИТЕКТУРЫ

## *Что такое SOA?*

SOA – это кульминационный этап развития различных стратегий интеграции на протяжении многих лет. С тех пор, как ИТ-индустрия нашла замену мейнфреймам, стали доминировать небольшие компьютеры. С ростом числа таких компьютеров всё острее вставала задача обеспечения возможности взаимодействия их между собой. Потребность заставить расположенные рядом компьютеры «говорить» друг с другом привела к созданию сначала клиент-серверной архитектуры, а затем – развитию распределённых архитектур, в которых данные и функции распределяются между компонентами системы.

## *Интеграция «точка к точке»*

В начале развития системной интеграции взаимодействие между клиентом и сервером по принципу «точка к точке» использовалось чаще всего. Клиент должен был знать о сервере, сервер – о клиенте, и, вдобавок, оба они должны были использовать одинаковый коммуникационный протокол. Если же протоколы различались, то каждый участник обмена должен был самостоятельно выполнять преобразования для того, чтобы его сообщения стали понятны другим участникам.

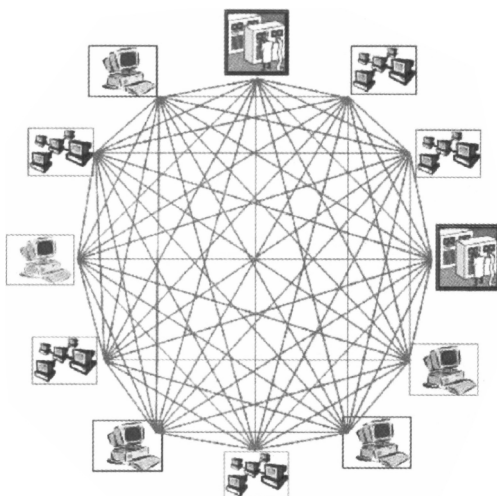
Применение клиент-серверной архитектуры и интеграции «точка к точке» во многих случаях привело к тому, что сейчас называется «спагетти-интеграция». На рис. 8 показано, насколько сложной может быть задача интеграции приложений компании по принципу «точка к точке». Для того, чтобы связать всего двенадцать приложений может потребоваться определить до шестидесяти шести различных интерфейсов. Каждая клиент-серверная система имеет собственный язык и модель данных. Если добавится новое приложение, то каждое из существующих должно будет взаимодействовать с ним через специальный дополнительный интерфейс. Каждый интерфейс применяет различные преобразования данных. Легко видеть, что обслуживание и поддержка подобной архитектуры может очень быстро стать крайне сложной. Особенно это касается производственного уровня (уровня 3). Интерфейсы промежуточного (middleware) программного обеспечения на этом уровне постоянно изменяются из-за ввода в производство новых продуктов, перехода от пилотного производства к массовому, прекращения выпуска старой продукции и непрерывного улучшения и развития производственных процессов.

## *Интеграция корпоративных приложений*

В конце 90-х годов потребность в упрощении архитектуры «точка к точке» привела к тому, что популярность приобрела новая концепция – интеграция корпоративных приложений (Enterprise Application Integration, EAI). EAI использует принцип интеграции «ступица и спицы» (hub-and-spoke), в которой все сообщения перед передачей адресату проходят через единый центр – концентратор. Концентратор отвечает за маршрутизацию, перевод с одного языка на другой, обеспечение правильной последовательности сообщений и преобразование содержимого и протокола сообщения перед тем, как сообщение будет передано конечному адресату. Подход EAI позволял значительно упростить внедрение и поддержку. Теперь отправителю сообщений нужно было уметь взаимодействовать только с концентратором, а не с большим количеством возможных партнёров. Вследствие этого программная логика, необходимая для маршрутизации и преобразования сообщений была перенесена из приложения в концентратор. Следовательно, отдельные приложения (те самые «спицы») теперь были должны реализовывать только те задачи, для которых они и предназначались, а не заниматься организацией связи с неопределённым количеством адресатов. Когда данные попадали в концентратор, они преобразовывались из формата, специфичного для данного приложения в некоторый общий канонический формат. При

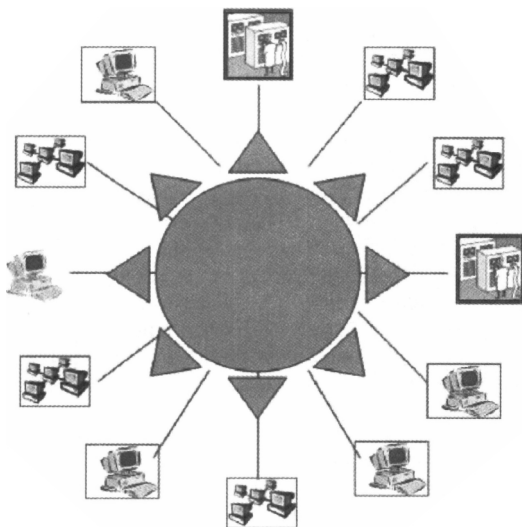
выходе данных из концентратора происходило новое преобразование, но теперь уже в формат другого приложения.

**Рис.8. Интеграция  
«точка-к-точке»**



Как видно из рис.9, подход EAI позволяет сократить количество интерфейсов с 66 до 12 (по числу приложений). Двенадцать приложений должны лишь уметь общаться с концентратором. Концентратор же должен знать, как взаимодействовать с любым из двенадцати приложений.

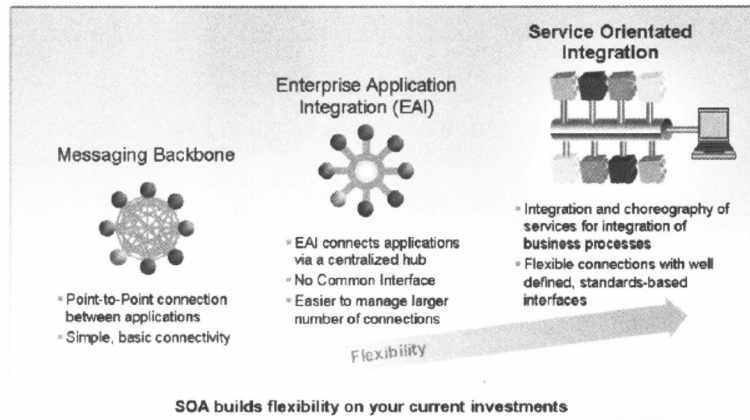
**Рис.9. Интеграция с  
использованием  
концентратора**



Однако даже при таком способе интеграции логика маршрутизации и преобразования данных может стать трудной для обслуживания без использования специальных методов управления конфигурацией. Рано или поздно появляется тенденция «переползания» бизнес-логики из приложений в концентратор, в результате чего концентратор становится всё труднее адаптировать для работы с новыми приложениями. Кроме того, с EAI всё равно приходится иметь дело с разнообразными крупными и монолитными приложениями, использующими разные способы коммуникации.

Несмотря на то, что EAI – большой шаг вперёд по сравнению с архитектурой «точка к точке», она всё ещё оставляет пробелы, которые заполняются только SOA в комплексе с соответствующими стандартами. На рис. 10 показано, как методы интеграции изменялись со временем. Сейчас наблюдается переход к подходу, основанному на сервисах, при котором маленькие приложения, способные предоставить информацию о своих возможностях и особенностях, взаимодействуют и обмениваются информацией посредством сервисной шины предприятия (ESB). ESB – один из основных компонентов сервис-ориентированной архитектуры.

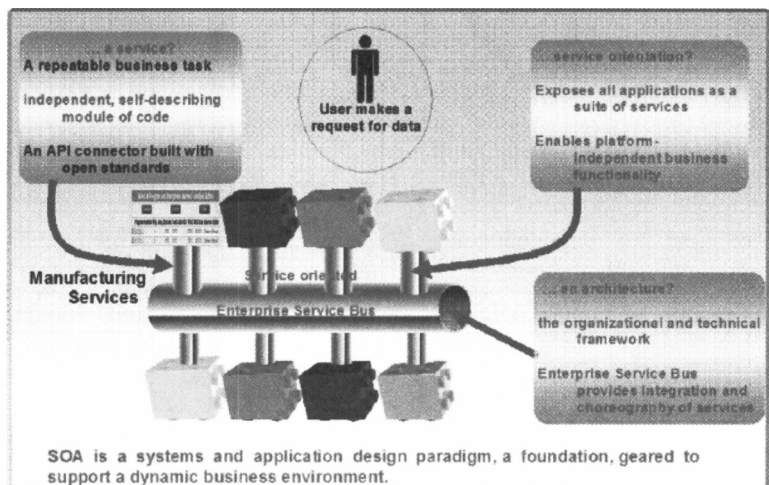
**Рис. 10. Эволюция подходов к интеграции**



### ***Сервис-ориентированная архитектура (SOA)***

SOA – модель информационной системы, предоставляющая компаниям новую степень свободы в разработке бизнес-приложений. SOA ориентируется не столько на интеграцию приложений, сколько на создание новых приложений из уже имеющихся. Архитектура делает возможным создания составных приложений из независимых, предоставляющих информацию о себе (с возможностью самоописания), взаимозаменяемых модулей, называемых сервисами. К сервисам можно обращаться посредством сервисной шины, они могут объединяться для формирования бизнес-процесса или составного приложения посредством координации процессов. Следовательно, основные компоненты SOA:

- сервисы;
- сервисная шина;
- координация процессов;
- преобразование и маршрутизация сообщений;
- реестр сервисов.



## Компоненты сервис-ориентированной архитектуры

### Сервисы

Сервисы – основной строительный блок в SOA. Довольно трудно выбрать точное определение сервиса. Чаще всего применяются следующие определения сервиса:

- повторяющаяся бизнес-задача;
- независимый программный модуль с возможностью самоописания;
- доступный ресурс, выполняющий повторяющееся действие и описанный с помощью открытой спецификации.

Вместо того чтобы пытаться найти точное определение, давайте рассмотрим основные концепции сервисов в SOA.

*В основу сервисов положены не ИТ-концепции, а концепции бизнеса.* Одна из основных идей SOA заключается в том, что бизнес управляет информационными технологиями, а не они диктуют бизнесу правила игры. Этим и достигается гибкость, одно из главнейших преимуществ использования SOA.

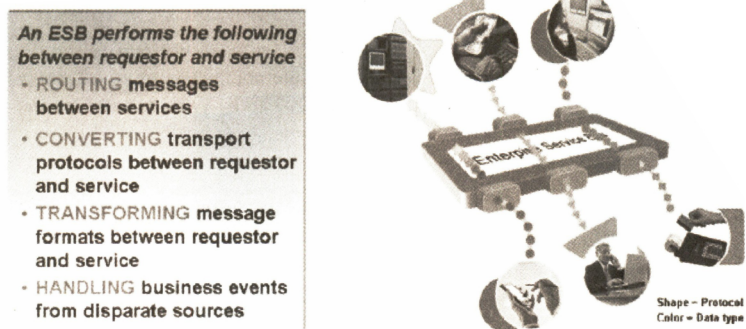
Сервисы способны к самоописанию, для которого применяется язык описания веб-сервисов (Web Services Description Language, WSDL). WSDL содержит средства для описания интерфейса сервиса, доступных операций, политик использования и т.д.

*Сервисы можно использовать повторно.* Искусство создания (выделения) сервисов состоит в выборе «гранулярности» нового сервиса. Степень гранулярности определяет возможность повторного использования сервиса.

*Сервисы самодостаточны и независимы.* Сервисы могут работать как сами по себе, безо всякого внешнего участия, так и комбинироваться с другими сервисами для создания составных приложений. Примеры обоих «режимов работы» приводятся в данном руководстве.

Сервисная шина предприятия (ESB) – гибкая инфраструктура для интеграции приложений и сервисов. Одна из целей применения ESB – уменьшить количество, размер и сложность интерфейсов в SOA. ESB – не программный продукт, это новая точка зрения на интеграцию приложений, координацию ресурсов и работу с информацией. ESB позволяет соединять приложения, написанные на разных языках и работающие на разных платформах.

**Рис.12. Что такое ESB**



## **Сервисная шина**

### **Как ESB решает проблему интеграции «точка к точке»?**

ESB предоставляет центральную точку для подключения распределённых приложений. Это не похоже на предыдущие подходы, такие как удалённый вызов процедур (RPC) или распределённые объекты. ESB использует шаблоны, позволяющие соединять приложения, работающие параллельно на разных платформах, написанные на разных языках программирования и применяющие разные программные модели.

## **Координация процессов**

Если рассматривать SOA как одну из стратегий интеграции, то она выглядит как каталог атомарных бизнес-сервисов с возможностью самоописания, используемый совместно для создания бизнес-процесса. Зачастую возникновение одного события (например, приход шасси на линию сборки) потребует начала взаимодействия между несколькими разными производственными и бизнес-системами. Для обеспечения такого взаимодействия в SOA применяется координация процессов.

Координация процессов позволяет комбинировать несколько бизнес-сервисов для реализации бизнес-процесса. В SOA координация реализуется с помощью BPEL (Business Process Execution Language, язык выполнения бизнес процессов). BPEL основан на XML, описание на BPEL компилируется в исполняемый код, что позволяет «собрать» бизнес-процесс из доступных сервисов. Более подробно BPEL описан в следующей главе.

В качестве примера бизнес-процесса можно привести процесс отслеживания автомобиля в процессе сборки. Когда автомобиль прибывает в зону сборки, формируется событие, которое запускает бизнес-процесс, использующий следующие сервисы:

- обновление информации о местонахождении автомобиля в MES;
- обновление записи об истории изготовления автомобиля в MES;
- получение информации об опциях из заказа;
- распространение информации об опциях в ПЛК, управляющие сборочными участками;
- передача сборочных инструкций на локальный принтер.

Процесс, описанный выше, можно назвать составным приложением. С помощью доступных сервисов из каталога можно собрать много составных приложений. В свою очередь, такое приложение можно рассматривать как единый, более сложный сервис и использовать в других бизнес-процессах. SOA делает возможным такой способ разработки (сборки) приложений. Теперь больше нет необходимости строить большие монолитные приложения и точно специфицировать интерфейсы прикладного программирования (API). SOA позволяет динамически создавать приложения путём соединения сервисов, работающих

поверх существующих или унаследованных приложений, либо новых сервис-ориентированных приложений.

Для выполнения бизнес-процесса, описанного на BPEL, применяется специальное программное обеспечение – среда исполнения процессов (process engine). IBM, BEA, Oracle, GE Fanuc, SAP и другие компании поставляют среды исполнения вместе со своими серверными продуктами. Исполняемые сервисы остаются доступными для использования в других бизнес-процессах.

### **Преобразование и маршрутизация сообщений**

В SOA сервисная шина ответственна за выполнение следующих задач классической интеграции корпоративных приложений:

- преобразование протоколов, например из JMS в SOAP и обратно;
- преобразование форматов сообщений, например, преобразование сообщения фиксированной структуры в OAG BOD;
- сопоставление сообщений, например, преобразование значений, понятных SAP, в значения, понятные системам производственного уровня;
- маршрутизация сообщений, когда, например, сообщение, полученное от системы производственного уровня, нужно преобразовать и передать в три другие системы;
- обеспечение безопасности, журналирование, поддержка транзакций.

### **Реестр сервисов**

В процессе внедрения SOA на предприятии появляется всё больше и больше сервисов, которые необходимо как-то идентифицировать. Со временем становится важным иметь центральное хранилище, необходимое для регистрации и управления сервисами, как это показано на рис. 13. Реестр сервисов может использоваться как для хранения описания интерфейсов сервисов (на WSDL), так и для хранения различных метаданных сервисов. Реестр может предоставлять средства контроля версий и управления изменениями. Его можно рассматривать как систему управления библиотеками при разработке с использованием SOA.

Один из примеров спецификации реестра – стандарт UDDI (Universal Description, Discovery and Integration, Универсальное описание, поиск и интеграция), определяющий возможности по хранению и извлечению информации о веб-сервисах. Программные продукты, реализующие реестр сервисов, выпускаются многими компаниями, в том числе Sun, Microsoft и IBM.

Реестр сервисов может использоваться для поиска, публикации, управления и подписки на сервисы. Реестр может использоваться также для хранения метаданных, таких как:

- провайдер сервиса;
- доступность;
- время, в которое сервис доступен для использования;
- характеристики производительности и показатели производительности (KPI).

### ***Жизненный цикл SOA: моделирование, реализация, выполнение, управление***

#### **Моделирование**

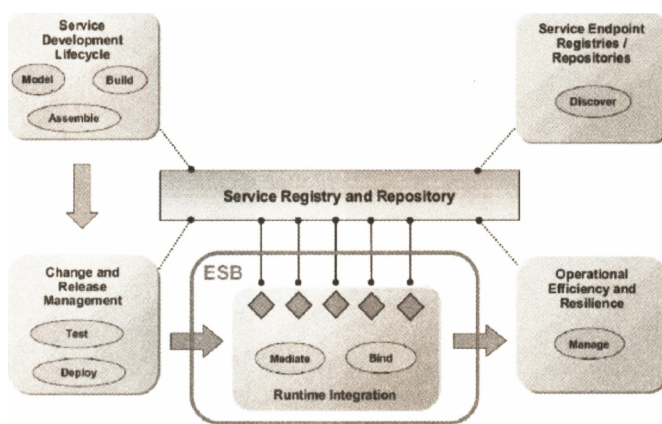
Когда SOA стала одной из ведущих архитектур распределённых систем, встала потребность в разработке модели программирования, способной помочь вендорам получить целостное представление о её возможностях. В разработке такой модели приняли участие IBM, BEA, Oracle, SAP AG и другие компании. В компонентной сервисной архитектуре (Service Component Architecture, SCA), основной части модели, бизнес-логика представляется



с помощью сервис-ориентированных интерфейсов. Доступ к функциональности других компонентов (ссылки на сервисы) выполняется через такие же интерфейсы. SCA разделяет разработку SOA-приложения на два этапа:

- реализация сервисных компонентов, которые предоставляют и используют сервисы;
- сборка компонентов путём привязки сервисов к реализациям.

**Рис.13. Пример реестра сервисов**



SCA использует минимальное количество API и реализуется на нескольких языках программирования, таких как Java, C++, COBOL, BPEL, XSLT, SQL и XQuery. C# в настоящий момент не поддерживается (если вы используете C#, вы можете обратить внимание на WCF). SCA поддерживает синхронно-асинхронную связь, связь «запрос-ответ», а также связь на основе сообщений. Привязка интерфейсов, также как и задание уровня обслуживания (QoS) выполняется декларативно и не зависит от реализации сервиса. Для привязки могут использоваться сообщения, веб-сервисы или CORBA IIOP. QoS отражает безопасность и надёжность передачи сообщений, а также возможность организации транзакций.

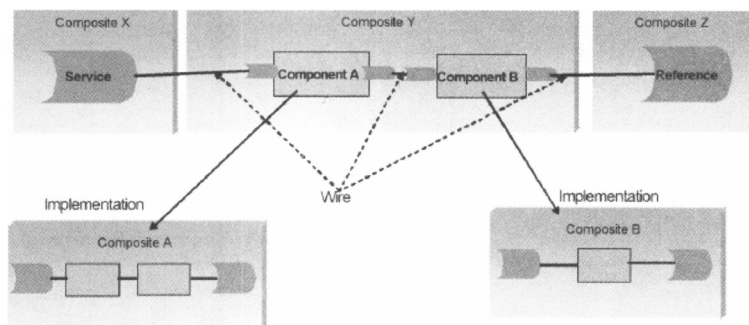
На рис.14 приведён пример SCA, компоненты которой (A и B) используют сервисы (входящие стрелки) и предоставляют интерфейс, с помощью которого их также можно использовать. Компоненты A и B связываются друг с другом для того, чтобы создать композитное приложение (Y), и конфигурируются путём отображения настроечных данных композитного приложения на настройки компонентов.

Сервисные объекты данных (Service Data Object, SDO) – ещё один элемент модели программирования, применяемый совместно с SCA. SDO даёт пользователю возможность получать доступ к данным, хранящимся в различных приложениях, и обрабатывать эти данные с помощью единого API. SDO описывают данные, передаваемые между сервисами SCA, и предоставляют общий интерфейс к данным аналогично тому, как SCA предоставляет общий интерфейс к сервисам. Как показано на рис.15, SDO можно представить в виде графа, узлы которого соответствуют объектам данных, каждый из которых содержит и метаданные, и значение. В SDO также хранится информация об изменениях объекта данных, сделанных до синхронизации с хранилищем данных.

SOA-системы строятся так, чтобы созданные пользователем SDO могли использоваться и распознаваться ESB на уровне преобразования данных с помощью специальных метаданных. Некоторые компании используют специальные посредники (SDO data mediators), предназначенные для создания SDO на основе существующих источников данных и последующего обновления источников данных в соответствии с изменениями, вносимыми

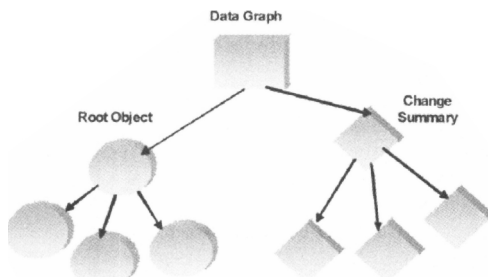
в SDO. В качестве примера таких посредников можно привести JDBC, XML и EJB. Как и можно было бы ожидать, посредники и SDO снижают затраты на программирование и предоставляют универсальный интерфейс к данным, не зависящий от источника данных.

**Рис.14. Пример сервисной компонентной архитектуры**



Другая важная черта модели SOA – возможность объединять компоненты SOA в более крупные составные приложения с использованием координации. Такой подход иногда называют «программирование–в-большом», общепринятым стандартом для решения этой задачи является язык BPEL, основанный на XML. BPEL поддерживает описание как коротких (микро), так и долговременных (макро) моделей жизненного цикла. Процесс, описанный на BPEL, вызывает сервисы в соответствии со стандартом WSDL 1.1. BPEL позволяет работать как с процессами, так и с управляющими данными, т.е. SDO. BPEL поддерживает циклы и ветвления, последовательную и параллельную обработку программы, возможность организации транзакций и компенсации транзакций. Поддержка транзакций реализована с использованием некоторых стандартов WS-I (например, WS-Coordination), описанных в следующих разделах руководства. Некоторые веб-сервисы, участвующие в процессе, могут не иметь возможности участвовать в двухфазной фиксации транзакции (2PC), и, следовательно, являются кандидатами для компенсации, которая реализуется как отдельный поток BPEL-процесса.

**Рис.15. Пример объекта данных сервиса**



Компоненты, вызываемые в BPEL-процессе, могут располагаться на локальных или удалённых системах и реализовываться на любом языке, поддерживающем веб-сервисы и

стандарты WSDL. Это делает BPEL отличным средством для интеграции и сборки функций, предоставляемых системами разных производителей, к тому же требующими разных навыков для работы с ними. Сами BPEL-процессы могут иметь собственный интерфейс и использоваться другими бизнес-процессами как обычный сервис, что позволяет скрывать сложность, инкапсулируя сложные бизнес-процессы в сервисы более высокого уровня, доступные на всём производстве.

### **Реализация**

Среды программирования дают возможность создания программных компонент, способных устанавливаться и работать на любых серверах заказчика. Большинство средств поддерживают либо декларативное объявление транзакций и веб-сервисов (.NET), либо мастера и конфигурационные параметры, позволяющие решить ту же задачу (J2EE с дескрипторами развёртывания). Для создания композитного приложения на BPEL в средах разработки имеются специальные инструменты, транслирующие описание последовательности сервисов (обычно графическое) в описание процесса на BPEL, понятное соответствующей среде исполнения. Стандарт BPEL довольно абстрактен и не содержит никаких требований по реализации выполнения на конкретном аппаратном и программном обеспечении. Каждый поставщик берёт описание BPEL и расширяет его для того, чтобы максимально использовать свойство среды исполнения. Именно поэтому необходимо применять инструменты для трансляции в BPEL, созданные специально для этой среды.

При использовании SOA на производственном предприятии в общей архитектуре присутствует «расширенная» ESB – сервисная шина производства (MSB). MSB добавляет к ранее описанным функциям ESB следующие функции:

- инструментальные средства (производственные инструментальные средства) для моделирования событий и реакции на них в контексте конфигурации производственной линии;
- сервисы доступа к оборудованию, расширяющие возможности связи производственного оборудования и ESB;
- стандартные производственные сервисы, реализующие основные функции MES (например, отслеживание незавершённого производства);
- поддержка стандартов интеграции производственных систем (например, ISA-95 и OAGIS).

Производственные инструментальные средства – ключевой элемент в обеспечении возможности настройки и конфигурации MSB пользователями, не являющимися программистами. С помощью этих средств можно создать конфигурацию производственной линии, включающую следующее:

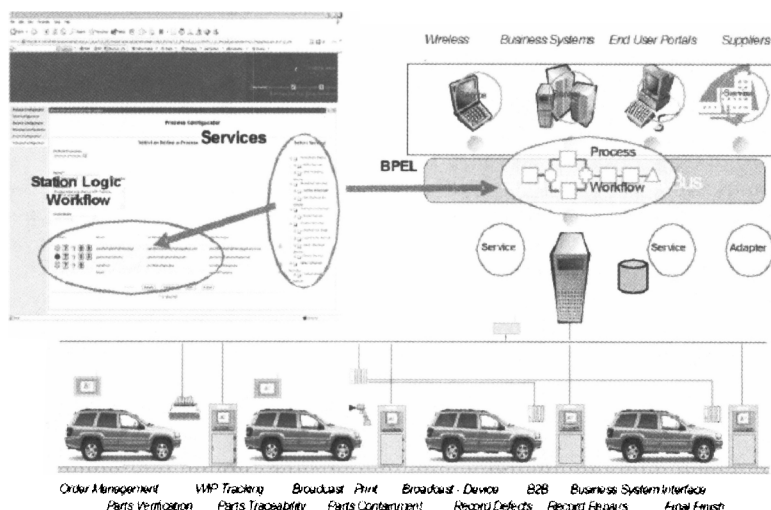
- общий вид производственной линии;
- бизнес-процессы, поддерживающие работу производственной линии;
- события, по которым начинается выполнение бизнес-процессов;
- взаимосвязь между участками линии и событиями;
- соответствие между данными в событиях и сервисами в бизнес-процессах, запускаемых этими событиями.

На рис.16 показан пример использования инструментальных средств для создания композитного приложения или бизнес-процесса из доступных сервисов. Инструментальные средства дают пользователю возможность собрать сервисы вместе и получить композитное приложение. Для того, чтобы приложение функционировало в соответствии с потоками работ, специфичных для предприятия, оно должно запускаться в ответ на события, информация о которых поступает в ESB.

После создания конфигурации производственной линии, инструментальные средства применяются для генерации и развёртывания различных артефактов (например, исполняемого кода и конфигурационных настроек), необходимых для реализации созданной

конфигурации. Обычная форма представления выходного результата – программа на BPEL, которая при необходимости может быть преобразована в описание потока работ для одной из поддерживаемых сред исполнения. После развёртывания в среде исполнения, конфигурация линии может использоваться для обработки событий.

**Рис.16.**  
**Конструировани**  
**е композитных**  
**приложений с**  
**использованием**  
**сервисов и BPEL**



## Исполнение

Ключевой компонент любой среды исполнения в SOA – сервисная шина предприятия (ESB). Информация о событиях поступает в ESB в самой разной форме, в качестве реакции на то или иное событие происходит запуск бизнес-процесса. Как было сказано выше, ESB ответственна за маршрутизацию, преобразование и обработку бизнес-событий. Также мы ввели понятие расширения ESB (MSB) для промышленных предприятий, в котором к основным функциям ESB добавляются функции, специально предназначенные для производственной среды.

Сервисы доступа к оборудованию получают и передают данные на уровень производственного оборудования (рис.17). С помощью сервисов доступа устройства посылают информацию о событиях ESB. ESB преобразует событие из исходного формата в общий канонический формат (внутренний формат), зависящий от конфигурации линии, заданной в производственных инструментальных средствах. После преобразования в канонический формат информация о событии может использоваться компонентами MSB для обращения к соответствующему бизнес-процессу, определяемому, опять же, конфигурацией линии. Результат работы бизнес-процесса преобразуется ESB в формат уровня оборудования и передается обратно на этот уровень с помощью сервисов доступа к оборудованию MSB.

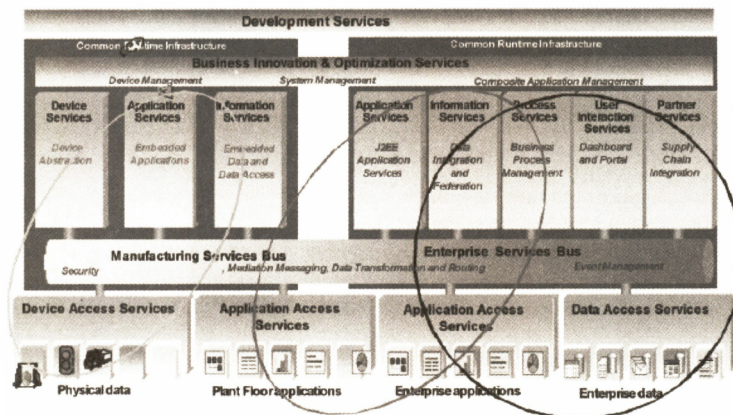
На рис.17 показана архитектура MSB. Важно отметить, что информация о событиях может поступать с любого уровня компании, однако способ обработки не зависит от источника события. Информация событию поступает в ESB в любом формате, поддерживаемом источником и затем преобразуется в канонический формат MSB с использованием метаданных конфигурации. После преобразования MSB выбирает нужный бизнес-процесс для выполнения путём просмотра, обработки и сравнения информации о событии с метаданными. После завершения выполнения бизнес-процесса его результат

преобразуется из канонического формата в исходный, поддерживаемый приложением, отправившим событие, и отправляется обратно.

Рис.17.  
Производственная  
сервисная шина

## Manufacturing Process Integration Architecture

Device to Plant Floor to Enterprise



Если изменяется бизнес-процесс, или, в более общем случае, конфигурация линии, то она может быть изменена в инструментальных средствах и повторно развернута в среде исполнения. Подобные изменения не требуют программирования и каких-либо других навыков создания приложений. Для обработки событий после этого будет использоваться новая конфигурация.

### Управление

Важнейшей задачей остаётся обеспечение возможности мониторинга и управления потоком событий. MSB ответственна за протоколирование всех событий, отслеживание времени реакции на событие и выполненных действие и расчёт показателей производительности. Все данные о событиях протоколируются с использованием общепринятых стандартов, например спецификации Common Based Events (CBEs, Обобщённые события). В настоящее время эта спецификация является частью стандарта OASIS WSDM: [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wsdm](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsdm). Применение стандартных форматов представления событий добавляет дополнительную гибкость, позволяя использовать любое средство мониторинга, поддерживающее эти стандарты.

После протоколирования события могут отображаться несколькими способами. Для представления данных на порталах можно воспользоваться технологией портлетов. Другие веб-технологии могут использоваться для визуализации информации о сообщениях в браузере. Также для отображения потоков событий могут применяться собственные программные средства поставщика решения, поддерживающие формат CBE.

## **Реестры и репозитории сервисов**

С ростом числа сервисов реестр сервисов становится важным инструментом разработки и управления, так как он обеспечивает доступность информации об имеющихся сервисах и делает возможным их повторное использование во всей компании.

Реестр используется и во время исполнения процесса. Например, для динамического выбора альтернативных сервисов (например, при недоступности провайдера) может использоваться реестр, поддерживающий WS-Addressing. Правда, в этом случае возможно снижение производительности, о чём необходимо помнить при реализации подобных систем на MES-уровне или уровне управления технологическим процессом, так как здесь требования к производительности очень строги.

## **ПРИМЕНЕНИЕ СТАНДАРТОВ ДЛЯ ДОСТИЖЕНИЯ МАКСИМАЛЬНОГО ЭФФЕКТА ОТ SOA**

Переход от интеграции «точка-к-точке» к использованию концентраторов и, затем, к SOA значительно повысил эффективность организации обмена данными между приложениями. Дополнительное преимущество заключается в том, что теперь приложения легко могут адаптироваться к изменениям в бизнесе. В разделе «Обзор SOA» подробно описаны разные способы интеграции. Несмотря на то, что применение концентраторов и SOA серьёзно облегчает интеграцию, возможность для дальнейшего улучшения всё равно остаётся. Стандарты могут существенно усилить эффект от интеграции. Ниже приведены примеры таких стандартов.

Стандартные коммуникационные протоколы позволяют легко связывать шины или концентраторы различных производителей.

Стандарты J2EE позволяют организовать совместную работу нескольких серверов приложений.

Стандарты на содержание (семантику) сообщений снижают число преобразований, необходимых при передаче сообщения по шине. Например, преобразование сообщения шиной к стандартному формату снижает общее число преобразований при передаче от одного приложения к другому. Применение канонического формата может рассматриваться как аналог применения стандарта, однако в том случае, если канонический формат является закрытым, для дальнейшей передачи необходимы новые преобразования. Выбор канонического формата из числа стандартных приводит к ситуации, когда для подключения нового приложения или программного шлюза не будет требоваться никаких дополнительных преобразований, при условии, что они поддерживают тот же стандарт. Применение стандарта снижает также потребность в преобразованиях вследствие изменения рынка или бизнес-модели. В том случае, если сами приложения поддерживают промышленные стандарты, уменьшается число преобразований, выполняемых шиной, и, как следствие, повышается производительность обмена информацией.

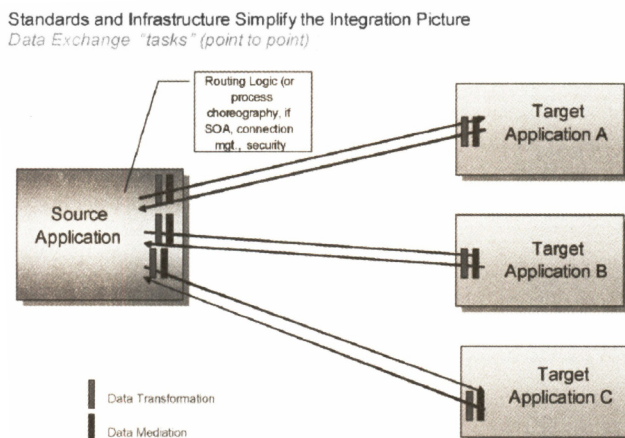
Число необходимых преобразований может быть столь большим, что использование ESB без соответствующих стандартов на содержание сообщения можно сравнить с интеграцией «точка-к-точке» с применением новой технологии. Рассмотрим основные операции, необходимые при обмене информацией между системами.

1. Выбор маршрута на основании правил маршрутизации и информации об адресате сообщения.
2. Преобразование данных из исходного формата в формат, поддерживаемый приложением-адресатом.
3. Выбор протокола передачи в зависимости от возможностей адресата.
4. Преобразование содержимого сообщения (например, перевод из килограмм в фунты).

Также в ходе обмена могут потребоваться операции, связанные с обеспечением безопасности, координацией (хореографией) процессов, мониторингом и т.д.

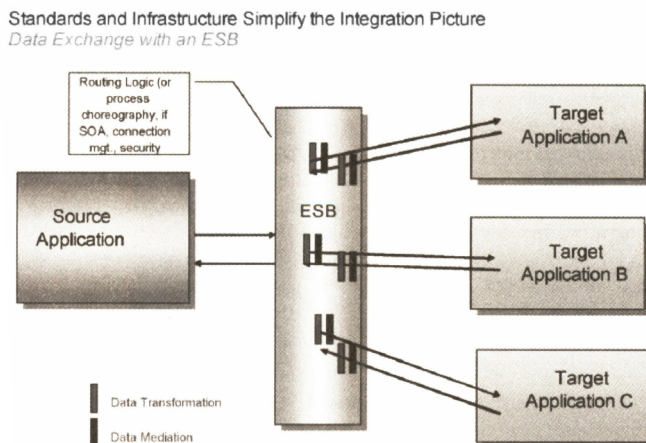
При использовании подхода «точка-к-точке» преобразование формата и содержимого сообщения требуется для каждого участника обмена (рис. 18).

**Рис. 18.**  
**Преобразования**  
**сообщений при**  
**обмене «точка-к-**  
**точке»**



Добавление шины сообщений упрощает архитектуру, так как маршрутизация, обеспечение безопасности и т.д. реализуются теперь в одном месте. Более простыми становятся и приложения-участники обмена, так как они не должны больше отвечать за преобразование формата и содержимого сообщений. Однако, как показано на рис. 19, эти преобразования всё равно имеют место в том случае, если при обмене не используются стандартные форматы сообщений.

**Рис. 19.**  
**Преобразования**  
**при обмене с**  
**использованием**  
**шины сообщений**

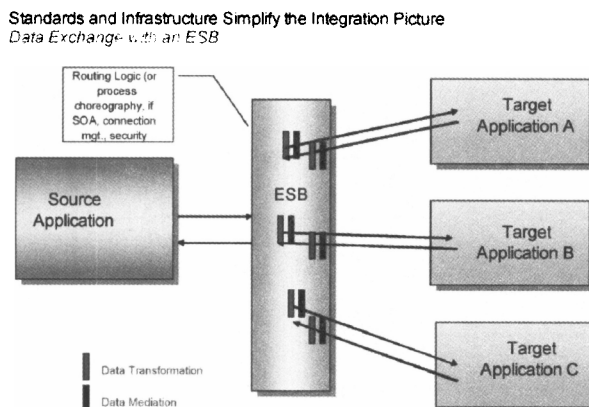


Если же все приложения будут придерживаться стандартов, то можно будет полностью отказаться от преобразований форматов и содержимого, что ещё больше упростит задачу

интеграции. Следует отметить, что конфигурируемое программное обеспечение промежуточного уровня (шина сообщений) делает возможным постепенный переход к использованию стандартных форматов сообщений.

На рис. 20 приведён пример такой ситуации. Только те приложения, которые не соответствуют стандартам, требуют преобразований форматов и данных.

**Рис. 20 Снижение количества преобразований при использовании стандартов**



Применение стандартов на формат и содержимое сообщений даёт возможность интеграции с программным обеспечением компаний-партнёров без необходимости в большом количестве преобразований, а также без особых проблем позволяет «подключать» коробочные программные продукты, также поддерживающие эти стандарты.

### **Стандарты на форматы и содержимое сообщений**

Независимо от того, какой подход к интеграции используется - SOA или EAI - применение при обмене общего канонического формата данных сокращает число преобразований сообщений, необходимых при взаимодействии нескольких приложений. Для систем, функционирующих на различных уровнях управления предприятием, разработано довольно много самых разных стандартов обмена данными. Такие стандарты могут служить отправной точкой в определении канонического формата, используемого ESB (или концентратором сообщений). На выбор стандарта серьёзно влияют индивидуальные особенности систем, в том числе и уровень управления, на котором работает система. Какой бы стандарт не был выбран, в большинстве случаев потребуется его адаптация к конкретной ситуации. Под адаптацией может пониматься как выбор некоторого подмножества стандарта, так и добавление к объектам, описанным в стандарте, необходимых атрибутов.

Некоторые стандарты, используемые на современных производственных предприятиях, показаны на рис. 21. Для более подробного описания этих стандартов обратитесь к Руководству MESA №26 (MESA White Paper 26 «Related Manufacturing Integration Standards, A Survey»).

### **XML**

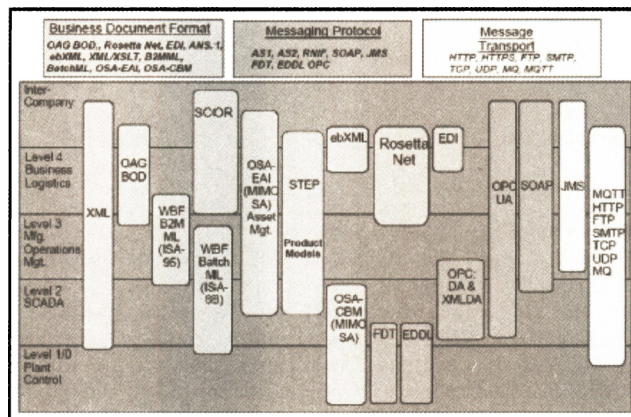
Расширяемый язык разметки (eXtensible Markup Language, XML) – открытый стандарт, определяющий язык разметки общего назначения. XML читаем как человеком, так и компьютерной программой. Этот язык служит основой для большинства стандартов, описываемых ниже.



## XML Schema

Язык XML Schema одобрен консорциумом W3C и может использоваться для определения форматов сообщений, используемых при обмене данными между MES, ERP и другими системами. XML Schema описывает структуру XML-документа и служит альтернативой для DTD (Document Type Definitions, определения типов документов). Такое описание формата и структуры документа называется XML Schema Definitions (XSD) и служит подобием контракта между создателем и потребителями документа.

**Рис. 21. Обзор стандартов интеграции для производства**



## ISA-95 и ISA-88

Появление сначала стандарта ISA-88 (управление рецептурным производством) в 1986 году, а затем и стандарта ISA-95 (IEC 62264) было обусловлено возросшей потребностью в интеграции корпоративных приложений и приложений, используемых при управлении производством. В частности, стандарт ISA-95 содержит:

- 1) функциональную и физическую модель иерархии приложений;
- 2) терминологию, применяемую для управления производственными операциями (Manufacturing Operations Management, MOM);
- 3) методологию описания обмена данными;
- 4) обобщённую модель данных;
- 5) модель деятельности для MOM.

Структура ISA-95 поясняется также на рис. 1.

## B2MML, BatchML

Язык разметки для обмена данными между корпоративными и производственными приложениями (Business to Manufacturing Markup Language, B2MML) – реализация концепций ISA-95 с использованием XML. Рецептурный язык разметки (Batch Markup Language, BatchML) – XML-реализация ISA-88. Оба языка разрабатываются и поддерживаются организацией WBF (ранее расшифровывалась как World Batch Forum – всемирная организация пользователей рецептурных производства).

## OAGIS

Спецификация интеграции Сообщества открытых приложений (Open Applications Group Integration Specification, OAGIS) определяет общий канонический стандарт на информацию, которой обмениваются приложения, а также на способы «упаковки» такой информации. OAGIS определяет так называемые «Документы бизнес-объектов» (Business Object Documents, BOD). Структура и заголовки BOD стандартны, а содержание специфично для каждого BOD. Как B2MML, так и BOD являются расширяемыми, правда, по-разному.

## **MIMOSA**

Открытая архитектура систем для интеграции корпоративных приложений (Open System Architecture for Enterprise Application Integration, OSA-EAI) – спецификация, разработанная Сообществом открытых систем управления производственной информацией (Machinery Information Management Open Systems Alliance). MIMOSA разрабатывает XML-спецификации для интеграции корпоративных приложений (EAI) и технического обслуживания по условию (Condition-based Maintenance). Такие спецификации включают детальные модели активов и оборудования производственного предприятия.

## **STEP**

Стандарт обмена данными о продукции (Standard for the Exchange of Product Model Data, STEP), или ISO 10303, определяет способы обмена цифровыми описаниями продукции между системами автоматизированного проектирования (САПР, CAD), системами автоматизированной разработки (CAE) и автоматизированного производства (CAM).

## **Сервисные платформы**

### **J2EE (Java EE)**

Java 2 Platform Enterprise Edition (J2EE) предоставляет стандартную платформу для программирования серверных приложений на Java. В состав J2EE входят такие спецификации, как JMS, веб-сервисы, XML, Enterprise JavaBeans, сервлеты, портлеты, Java Server Pages (JSP). J2EE даёт возможность создания и развёртывания переносимых, масштабируемых, отказоустойчивых, распределённых многозвенных приложений.

Первоначальная спецификация J2EE была разработана компанией Sun Microsystems. Последующие версии разрабатываются в рамках Java Community Process. Заметим, что, начиная с версии 1.5, название спецификации изменилось на Java Platform Enterprise Edition (Java EE).

### **.NET и интеграция приложений Microsoft**

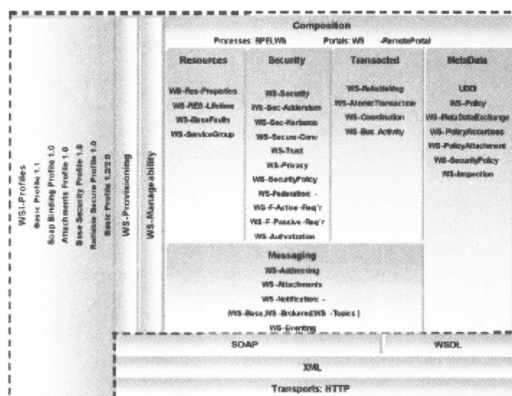
Одна из целей SOA состоит в предоставлении возможности интеграции и использования функциональности существующих приложений без затрат на переписывание и изменение кода. Для некоторых компаний важно также иметь возможность использовать сотрудников с навыками работы с различными технологиями. SOA и веб-сервисы позволяют достичь обе цели. Один из примеров – взаимодействие между приложениями для .NET и J2EE, и, в частности, взаимодействие между веб-сервисами на этих платформах. Как говорилось выше, несмотря на различия в реализации концепций SOA, обе технологии поддерживают ряд стандартов, что делает возможным взаимодействие между приложениями, реализованными на их основе. Некоторые из таких общепризнанных стандартов описаны ниже.

### **Взаимодействие между .NET и J2EE**

Интеграция приложений J2EE и .NET может производиться на одном из нескольких уровней. Разные архитектуры и шаблоны интеграции описаны в таких руководствах, как «Application Interoperability: Microsoft .NET and J2ee» ([http://download.microsoft.com/download/7/2/6/7269f183-639a-4e99-bd84-cc3e6515af86/PnP\\_J2EE\\_Interop\\_V1.pdf](http://download.microsoft.com/download/7/2/6/7269f183-639a-4e99-bd84-cc3e6515af86/PnP_J2EE_Interop_V1.pdf)) и «WebSphere and .NET Coexistence» (<http://www.redbooks.ibm.com/abstracts/sg247027.html?Open>). В руководствах рассматривается интеграция на основе веб-сервисов. Возможность взаимодействия на уровне веб-сервисов зависит от соответствия используемых технологий стандартам. Организация WS-I разработала набор стандартов, описывающих различные аспекты взаимодействия веб-сервисов. Степень, в которой технологии соответствуют этим стандартам, определяет и способность приложений к интеграции. Там, где поставщик программного обеспечения свободен в интерпретации требований стандартов, резко возрастает риск возникновения

### Профили WS-I (взаимодействие веб-сервисов)

**Рис. 22.**  
**Спецификации**  
**веб-сервисов для**  
**взаимодействия**  
**.NET и J2EE**



Базовый профиль WS-I (1.0) (<http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html>) – набор открытых спецификаций веб-сервисов. В спецификациях предъявляются требования к:

- 1) сообщениям – элементам протоколов обмена (сообщения HTTP/SOAP);
- 2) описаниям – типам, сообщениям и интерфейсам (WSDL);
- 3) регистрационным данным – регистрации и обнаружению сервисов (UDDI).

Базовый профиль требует соответствия определённым версиям стандартов – WSDL 1.1, SOAP 1.1, SSL 3.0 и т.д. Соответствия требованиям базового профиля в большинстве случаев

достаточно для беспрепятственного взаимодействия между приложениями .NET и J2EE. Ниже приведено несколько советов, позволяющих приблизить успех.

1. Лучше всего использовать документы/литералы, определённые WSDL.
2. Разработчики должны стараться обеспечивать уникальность сервисов в приложении, а при невозможности – уникальность полного доменного имени в пространстве имён.
3. В .NET и J2EE по-разному обрабатываются массивы. Например, в IBM WebSphere Application Server возможен возврат пустого значения (null) в качестве результата вызова веб-сервиса, что может привести к проблемам в приложениях на .NET. Поэтому для облегчения взаимодействия предлагается возвращать пустой массив вместо null.
4. При именовании сервисов следует следовать соглашениям Java и не использовать специальные символы наподобие подчёркивания (\_).
5. Следует, по возможности, избегать использования сложных типов данных. Некоторые из таких типов данных, а также соответствия между типами J2EE и .NET, приведены в приложении Е.

### **Безопасный профиль**

В безопасном профиле WS-I (<http://www.ws-i.org/Profiles/BasicSecurityProfile-1.0-2004-05-12.html>) содержатся требования к аутентификации, авторизации, обеспечению целостности сообщений и конфиденциальности. При аутентификации предлагается использовать имя пользователя, пароль в открытом виде или результат преобразования пароля (дайджест). Криптографические протоколы TLS (Transport Layer Security) и его предшественник SSL (Secure Sockets Layer) имеют широко известные прорехи в безопасности, учтённые в WS-I Security.

1. TLS не допускает выборочное шифрование сообщения, что может привести к значительному снижению производительности.
2. TLS гарантирует безопасность сообщения только во время передачи, поэтому, например, при ожидании в очереди сообщение не шифруется.

WS-I Security учитывает эти особенности. Производится шифрование только значимых частей сообщения, а границы шифрования отодвигаются к приложению. Выборочное шифрование сообщения позволяет избежать чрезмерной нагрузки, при этом информация для маршрутизации не шифруется, что позволяет проходить через сетевые экраны.

### **Надёжный безопасный профиль**

WS-ReliableMessaging (([http://docs.oasis-open.org/wsrn/ws-reliability/v1.1/wsrn-ws\\_reliability-1.1-spec-os.pdf](http://docs.oasis-open.org/wsrn/ws-reliability/v1.1/wsrn-ws_reliability-1.1-spec-os.pdf)) – спецификация надёжного обмена сообщениями между приложениями, поддерживающая четыре типа поведения:

- 1) сообщения доставляются только один раз;
- 2) сообщения доставляются как минимум один раз;
- 3) сообщения доставляются не более одного раза;
- 4) сообщения доставляются в порядке отправки.

В настоящее время стандарт WS-ReliableMessaging уже утверждён (<http://www.oasis-open.org/news/oasis-news-2007-06-21.php>). SOAP/JMS и SOAP/MQ – нестандартные реализации, создатели которых пытались обеспечить надёжный обмен сообщениями. Однако в силу того, что используются специфичные решения, SOAP/JMS и SOAP/MQ не обеспечивают возможности взаимодействия со всеми системами.

### **Дополнительные спецификации веб-сервисов**

#### **WS-Addressing**

Ещё один профиль, WS-Addressing (<http://www.w3.org/TR/2005/CR-ws-addrcore-20050817/>), был предложен в качестве стандарта Microsoft, IBM, BEA и другими и недавно

утверждён W3C. Он предлагает два конструктивных элемента – ссылки на начальную и конечную точки и информационные заголовки сообщений. Эти конструкции позволяют при необходимости динамически изменять точки обмена (например, в случае недоступности поставщика информации), а также обеспечить асинхронные коммуникации с сохранением состояний. Большинство запросов веб-сервисов представляют собой последовательность запрос-ответ или посылку, не требующую ожидания ответа. С помощью WS-Addressing разработчики могут реализовывать длительные асинхронные запросы, при которых сообщения перенаправляются в соответствии с информационными заголовками. Информационные заголовки позволяют получателю асинхронного ответа сопоставить сообщение исходному запросу и обеспечить сохранение состояний в случае двух и более запросов клиента к серверу.

### **WS-AT**

Спецификация WS-AT, Atomic Transactions (<http://schemas.xmlsoap.org/ws/2004/10/wsat/>) описывает координацию нескольких вызовов веб-сервисов в однородной или неоднородной среде для обеспечения их атомарной работы. Эта мощная возможность необходима, когда требуется координировать определённое число веб-сервисов в рамках одного потока работ, как это делается в BPEL. Например, при отказе одного сервиса может потребоваться откатить результаты действия предыдущего сервиса в потоке работ, который, например, выполнял модификацию базы данных. Координатор транзакций, использующий двухфазное подтверждение (2PC) может быть реализован как на .NET, так и на J2EE. Из-за особенностей 2PC следует использовать только для непродолжительных процессов. В J2EE для того, чтобы веб-сервис мог участвовать в транзакциях WS-AT, нужно изменить только дескриптор развёртывания. Изменений кода при этом не требуется. Для задания объектов, участвующих в транзакции, в .NET также используются декларативные описания, с помощью которых можно определить веб-сервисы, участвующие в глобальной транзакции.

### **Поддержка транзакций**

Существует две хорошо известные спецификации, определяющие реализацию поддержки транзакций в потоках работ, составленных из веб-сервисов. Спецификация WS-AT затрагивает координацию потока работ (или веб-сервиса) в пределах так называемой атомарной транзакции. Напротив, спецификация WS-BA (Business Agreement) посвящена длительным транзакциям (приостанавливаемым потокам работ). Две спецификации имеют общее название WSTransactions и расширяют спецификацию WS-Coordination, содержащую описание объединения нескольких веб-сервисов в один поток работ. Детальное описание WS-Coordination можно найти по адресу: <http://docs.oasis-open.org/ws-tx/wscoor/2006/06>.

Профили WS-I содержат детальные спецификации, позволяющие самому разнообразному набору программных продуктов от самых разных поставщиков обеспечить надёжный обмен информацией. При этом результат зависит только от того, насколько чётко приведена спецификация и насколько старательно поставщик выполнял требования стандартов. В тех случаях, когда стандарт содержит туманные требования или допускает несколько интерпретаций, реализация взаимодействия становится проблемой. Профили WS-I всё ещё находятся в разработке, и по мере того, как они становятся более зрелыми, можно ожидать радикального упрощения взаимодействия между системами. Сейчас модно говорить, что SOA – нечто большее, чем просто технология, но нельзя отрицать, что WS-I и взаимодействие между .NET и J2EE (не стоит забывать и о том, что WS-I легко реализовать и в других средах) – важнейшие направления развития информатики. Результаты же делают реализацию SOA всё более и более выгодной.

### **BPEL**

Спецификация языка исполнения бизнес-процессов для веб-сервисов (Business-process Execution Language for Web Services, WS-BPEL 2.0) предоставляет средства для формального описания бизнес-процессов и их взаимодействия с веб-сервисами и другими процессами.

WS-BPEL расширяет модель взаимодействия веб-сервисов и даёт возможность поддержки бизнес-транзакций. В разработку стандарта WS-BPEL внесли вклад такие компании как:

- IBM;
- BEA Systems;
- Microsoft;
- SAP AG;
- Siebel Systems.

WS-BPEL обеспечивает координацию сервисов для образования бизнес-процесса. Сервисы могут быть как локальными, так и удалёнными, могут реализовываться с использованием различных языков, что даёт возможность взаимодействия между сервисами в неоднородном окружении. Например, для унаследованных приложений может быть написана «обёртка» для включения в бизнес-процесс, описанный на BPEL. Слабая связность позволяет обновлять или менять сервис, не затрагивая весь бизнес-процесс.

Бизнес-процессы, описанные на BPEL, могут быть представлены в виде сервисов и, в свою очередь, использоваться как сервисы другими бизнес-процессами. WS-BPEL полностью совместим со стандартом J2EE. Расширения BPEL позволяют оператору влиять на ход бизнес-процесса.

### **OPC UA**

OPC UA (Unified Architecture, Унифицированная архитектура) – новый стандарт, предложенный OPC Foundation. OPC UA нацелена на решение проблемы интеграции в масштабах всего предприятия и основана на SOA, веб-сервисах, XML-архитектуре и WS-\* стандартах. Несмотря на применение стандартов, возможно использование закрытой технологии OPC для передачи данных (по соображениям производительности). В настоящее время стандарты OPC описывают разные объекты и методы в зависимости от типа сервера:

- Alarms and Events (сообщения);
- Data Access (доступ к данным);
- Historical Data Access (доступ к историческим данным);
- Commands (команды);
- Complex Data (составные данные).

Например, для каждого сервера определён метод «Browse», но синтаксис для каждого сервера является своим. OPC UA описывает унифицированную объектную модель, предоставляющую общий набор переменных, методов и событий, покрывая все модели существующих серверов. Предполагается поддержка .Net, Java, C/C++.

За более подробной информацией следует обратиться по адресу <http://www.opcfoundation.org>.

## ЗАКЛЮЧЕНИЕ

Несмотря на наличие технологий, поддерживающих использование SOA на производственном предприятии, большинство производителей не очень далеко продвинулись по этому пути. Преимущества перехода на SOA общепризнанны, во многих компаниях предпринимаются действия по реализации такого перехода, но это скорее эволюционный, чем революционный процесс.

В силу того, что SOA изначально ориентирована на использование стандартов, можно совместно использовать решений от нескольких поставщиков. При этом следует иметь в виду, что, как и в случае других стандартных технологий, здесь имеется некоторое пространство для манёвра, поэтому не всегда получается достичь полной совместимости различных решений. Например, в силу некоторой неопределённости стандарта на язык BPEL (WS-BPEL 2.0) нельзя гарантировать, что описание процесса, созданное с помощью программного обеспечения одного производителя может быть запущено на BPEL-движке другого производителя без необходимости каких-нибудь доработок. С веб-сервисами дело обстоит несколько лучше, в основном решения разных производителей совместимы между собой. Можно надеяться, что под давлением со стороны пользователей будут решены и остальные проблемы совместимости.

Производственные компании могут переходить на SOA уже сейчас, и многие уже начали этот переход. Как уже говорилось, не обязательно делать всё за один подход. Переход на SOA может выполняться в ходе проекта, при этом очень важно сразу определить архитектуру, которую нужно получить – тогда появится возможность планировать этапы процесса. Нужно сразу принять основные решения по архитектуре SOA и по мере выполнения проекта при необходимости корректировать эти решения.

Реальна ли SOA? Как видно из данного руководства, уже сейчас можно привести примеры реальных проектов, в ходе которых предприятия используют SOA, чтобы придать большую гибкость производственным системам. Так что на этот вопрос можно с уверенностью ответить «Да!».

## Приложение А. Примеры успешного использования SOA

В приведённых ниже двух примерах описываются детали реальных внедрений концепций SOA. Данными примерами список успешных внедрений не исчерпывается. При желании можно привести множество других примеров успешных проектов по переходу на SOA.

### *IBM Microelectronics Division*

В 2003 году подразделение IBM Global Engineering Solutions (GES) (включает Microelectronics Division и Engineering & Technology Services), являющееся частью Sever & Technology Group (STG) обратилась в департамент IT с просьбой обеспечить поддержку новой бизнес-модели. В частности, необходимо было обеспечить поддержку производства компонентов поставщиками азиатско-тихоокеанского региона (Asia Pacific, AP) для расширения производственной базы IBM. Проблема состояла в том, что новые производственные мощности должны использовать те же структуры данных, что и производственные подразделения IBM. Проще говоря, для разработчиков не должно быть отличий между оутсорсинговыми производствами и собственными производствами IBM, они должны получать одну и те же данные о ходе производства и логистическую информацию отовсюду. Проблема усугублялась наличием значительных объёмов данных, большая часть которых хранилась в закрытых форматах.

Для решения было предложено использовать SOA – создать систему с архитектурой, основанной на передаче сообщений, предоставляющую нужные сервисы потребителям. Сервисы, работающие на предприятиях внешних поставщиков должны предоставлять те же данные, что и сервисы, работающие на собственных предприятиях IBM. Так называемая «фабрика в коробке» - выделенный сервер – предоставляется поставщику и содержит всё необходимое для поддержки конкретного производства. Для того, чтобы связать эти удалённые серверы с концентратором в IBM выбрана методология MQ Series. Для преобразования и маршрутизации сообщений по безопасным каналам передачи выбран WebSphere Message Broker. Последняя часть решения – спецификация зоны безопасности (Demilitarized Zone, DMZ), отделённой сетевыми экранами от интернета и доверенных сетей предприятия для всех участников обмена. Внутри этой зоны должен находиться тот самый выделенный сервер. Тем самым обеспечивается соответствие требованиям безопасности как IBM, так и компании-поставщика.

Сжатые сроки, выделенные на решение задачи, не позволили выполнить пилотное внедрение, всё должно было работать сразу после поставки. Существовал серьёзный риск, связанный с необходимостью разбираться в унаследованных системах, однако опыт и навыки команды разработчиков, а также чёткое представление цели позволили его избежать.

Со времени первого развёртывания решение было модифицировано и расширено и в настоящий момент поддерживает 14000 логистических транзакций в день, сопровождаемых пересылкой нескольких гигабайт необработанных данных. При этом серьёзная поломка случилась лишь однажды, когда в результате выхода из строя подсистемы хранения сервера, расположенного у поставщика. Причиной стало то, что подсистемой хранения не использовался RAID 5. Сервис был недоступен для поставщика в течение восьми часов.

Определено, что решение является масштабируемым и задействует только 20% от своих возможностей. В оставшейся части данного раздела будет рассказано об основных деталях разработанной архитектуры, а также об эталонной архитектуре, разработанной в результате внедрения.



## **Моделирование процессов**

В начале проекта, получившего внутреннее название «Основа виртуального производства» (Virtual Factory Framework, VMF), было необходимо создать модель существующих бизнес-процессов, связанных с производством. Информация о таких процессах была фрагментарной, документация зачастую оказывалась устаревшей. Общая точка зрения на бизнес-процессы отсутствовала, не все владельцы процессов были идентифицированы (если были идентифицированы вообще). Вместе со сжатыми временными сроками всё это существенно повышало требования к уровню знаний о бизнесе членов команды разработки. Однако высокая квалификация разработчиков помогла сократить время до выпуска первой версии модели.

После выпуска первой версии разработчики вернулись назад и продолжили работу по созданию формализованного подхода и обобщению архитектуры для создания эталонной модели. В результате была разработана эталонная архитектура.

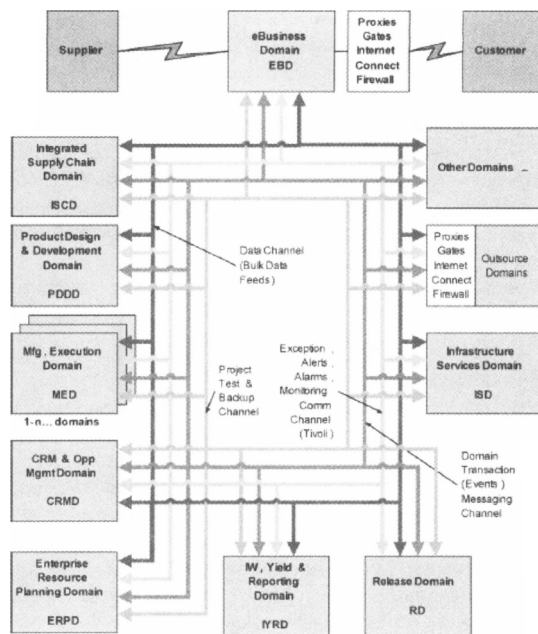
## **Эталонная архитектура IBM GES**

На схеме эталонной архитектуры, приведённой на рис. 23, показаны домены (зоны), на которые разбито Microelectronics Division. Каждый домен представляет собой самодостаточный бизнес-элемент со своим собственным набором приложений. При определении границ доменов использовалось единое правило – границы домена должны являться также границами для оутсорсинга.

Как видно из рисунка, каждый домен отделён от других шиной сообщений. Для транзакций, пересылки больших объёмов данных, а также аварийных сообщений используются отдельные шины. Для тестирования взаимодействия с системой нового программного обеспечения перед развёртыванием используется дополнительная шина транзакций. Все шины, за исключением шины мониторинга, реализованы на основе транзакций MQ, шина мониторинга использует протоколы Tivoli. Шина транзакций использует XML, протокол обмена и содержание сообщений реализуются в соответствии со спецификацией RosettaNet. При правильном использовании программных компонент для подключения (использования прокси) инфраструктурой передачи сообщений смогут пользоваться и поставщики.

Реализация эталонной архитектуры требует использования интерфейсных компонентов, особенно таких, которые могли бы обеспечить передачу, преобразование и маршрутизацию сообщений внутри зон. Подобных компонентов, удовлетворяющих всем требованиям, не существовало, поэтому было принято решение о разработке собственных. В ходе разработки была создана система, называемая MDI. Первоначально аббревиатура MDI расшифровывалась как Manufacturing Data Integrator (интегратор производственных данных), однако в дальнейшем система была переименована в Multi Data Source Integrator (интегратор данных из различных источников). MDI реализует концепции SOA и содержит собственные сервисы. Существующие системы могут «обёртываться» сервисами MDI, могут создаваться новые сервисы. Со временем планируется полный переход на MDI и SOA. Каждый MDI можно рассматривать как мини-ESB.

**Рис. 23.**  
**Эталонная**  
**архитектура v.1.2**  
**(2003, 2006)**



### **MDI, каркас для внедрения SOA**

На следующем рисунке показана архитектура MDI.

MDI реализуется на основе IBM WebSphere Business Integrator, WebSphere Message Broker и WebSphere MQ. Сервисы реализуются на WebSphere Application Server и IBM DB2. Для организации реестра сервисов используется IBM WebSphere Registry and Repository. Платформа MDI позволяет разработчикам существующих приложений портировать их в два компонента MDI: бизнес-правила, обрабатываемые WebSphere Business Integrator, и сервисы, реализованные на WebSphere Application Server. Сервисы, входящие в состав MDI, удовлетворяют требованиям к переносимости, сформулированным в спецификации SOAP. MDI также предоставляет графический интерфейс пользователя для доступа к части сервисов, которые могут вызываться вручную.

Для поддержки потоков логистических данных используется стандарт RosettaNet, расширенный IBM для работы не только с внешними, но и с внутренними поставщиками.

Кроме сервисов общего назначения доступно несколько сервисов, специфичных для полупроводниковой промышленности. На следующей диаграмме показаны основные компоненты и сервис, используемые в MDI. Заметим, что компонент Tool Control Infrastructure Operations (TCIOps) – закрытый компонент для обработки и преобразования сырых данных, позволяющий получить данные в стандартном формате.

Рис. 24. MDI

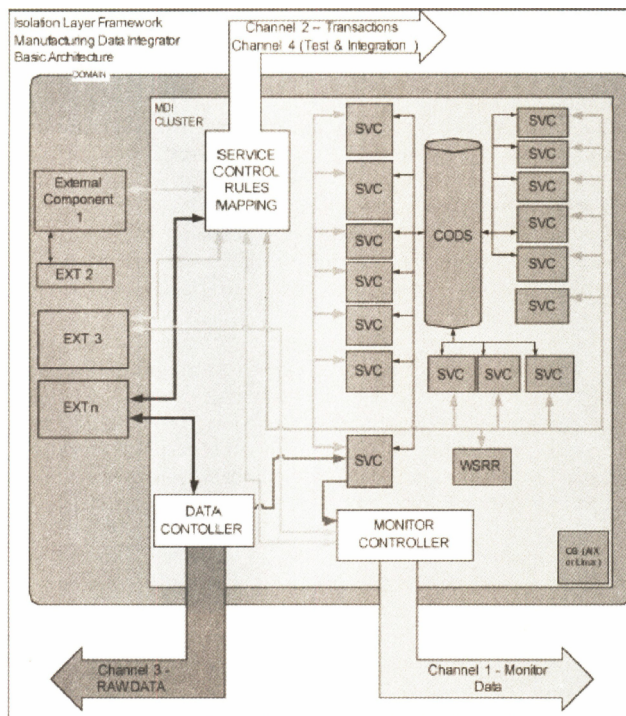
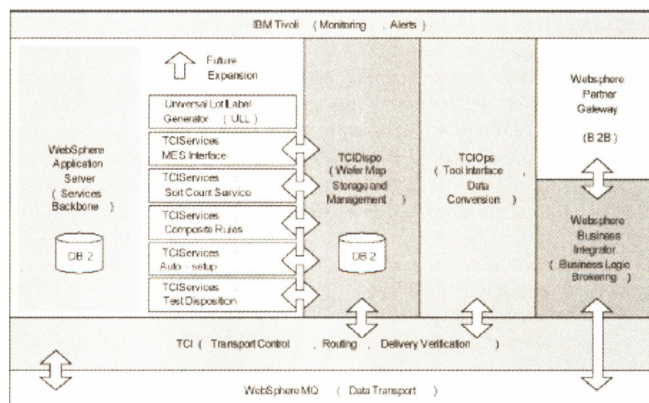


Рис. 25. Карта компонент MDI



### Основные результаты внедрения

Не один пример нельзя считать завершённым до тех пор, пока не будут получены ответы на такие вопросы, как «К чему всё это привело?», «Что бы вы изменили, если могли?», «Какие планы по дальнейшему развитию?».

Что касается первого вопроса, «К чему всё это привело», то были получены даже лучшие результаты, чем ожидалось. Спустя четыре года MD работает с гораздо большим

количеством поставщиков, чем ожидалось. Системы MDI остаются полностью жизнеспособными и обеспечивают выполнение даже тех задач, которые было трудно предвидеть (например, перегруппировка B2B-транзакций). Одна из главных «побед» была одержана при продаже одного из производственных предприятий, относящегося к домену, информационные потоки которого поддерживались MDI. Во время продажи MDI был переконфигурирован для маршрутизации входящих сигналов о ходе производства, ранее направлявшихся напрямую в MES. Теперь эти сигналы перенаправляются в SAP IDOC и попадают в SAP-систему поставщика. В ходе такой замены произошёл единственный четырёхчасовой перерыв в производстве в тот момент, когда IBM передавала управление поставщику. Важно отметить, что MES нельзя рассматривать в качестве интеграционного концентратора, а MDI – можно. Отделение интеграционных компонентов от приложения – важный шаг во внедрении SOA.

Второй вопрос – «Что бы вы изменили?». Первоначальная концепция предусматривала отдельный MDI для каждого домена даже в том случае, когда один домен является точной копией другого. По прошествии времени видно, что MDI может использоваться для обеспечения деятельности нескольких похожих доменов. Фактически, один MDI может использоваться для всех заводов одного поставщика при условии, что обеспечено соответствие спецификациям. При этом нужно учитывать риск возникновения сбоя – среднее время между сбоями должно быть большим даже без использования специальных технологий.

В ходе первого внедрения не использовалось достаточного количества мониторов, предполагалась надёжность и достаточность мониторов WebSphere MQ. В последствии была разработана так называемая «штрафная скамейка», обеспечивающая реализацию сервисов, о которых часто забывают. При исполнении потока работ сервисы обычно вызываются последовательно, при этом каждый из сервисов потребляет или преобразует информацию. При возникновении ошибки желательно было бы повторно сформировать стартовые сообщения для того, чтобы перезапустить поток. В подобных случаях и используется «скамейка штрафников» - специальный компонент, который может хранить, переупорядочивать и перезапускать транзакции. Как только возникает ошибка, сообщение немедленно направляется на «скамейку», где имеются все средства для обработки ошибки (обычно исправление причин исключительной ситуации и перезапуска транзакции). Такую возможность сложно переоценить при работе с пакетными XML-транзакциями, при которых сообщения зачастую приходят в произвольном порядке. «Штрафная скамейка» сохранит информацию о транзакциях, выполнит их упорядочивание и затем перезапуск. Всё это будет выполнено автоматически. Главный урок, усвоенный разработчиками, состоит в том, что автоматическая обработка ошибок крайне важна и её возможность следует закладывать ещё на этапе проектирования.

В процессе первого развёртывания системы возникла ситуация, чуть было не приведшая к остановке всего проекта. Ситуация была связана со сложностью конфигурирования WebSphere MQ в глобальном масштабе. Добавление прокси в демилитаризованных зонах и избыточных путей передачи сообщений значительно усложнили тестирование. В конце концов была реализована первоначальная архитектура, однако в ходе реализации не раз приходилось обращаться к самому высокому уровню службы поддержки WebSphere MQ. Для облегчения дальнейшей работы были определены несколько типичных вариантов конфигурации.

Также в ходе внедрения была обнаружена проблема, связанная с порядком прихода сообщений. В качестве причин возникновения проблемы было выявлено несколько вариантов настройки приоритетов и персистентности сообщений при совместной работе WebSphere MQ и JMS. В результате усвоен ещё один урок – если две технологии предоставляют возможность приоритизации потоков, то следует выбрать одну из них и в дальнейшем придерживаться сделанного выбора. В данном случае выбор был сделан в

пользу использования приоритетов WebSphere MQ, так как в дальнейшем ожидается работа с потоками, не использующими JMS.

Ещё одно интересное наблюдение связано с широко распространённым мнением о том, что декомпозиция приложения на сервисы «отвязывает» бизнес-логику от программного обеспечения и требует от команды разработчиков глубоких познаний в бизнесе. Предполагалось, что перенос бизнес-правил в WebSphere Business Integrator приведёт к постепенному отказу от услуг ИТ-служб. Однако оказалось, что моделирование бизнес-процессов всё ещё относится к сфере информационных технологий и непохоже, что ситуация когда-нибудь изменится. Единственное изменение заключалось в большем использовании другой группы ИТ-специалистов (тех, кто занимается проектированием и созданием потоков работ). Такой подход позволяет увеличивать глобальное присутствие, и, в то же время, держать знания о процессах при себе. С развитием средств создания автоматизированных сервисов, языка BPEL и т.д. на первый план выйдут новые специалисты в области ИТ – бизнес-архитекторы и аналитики.

Первоначально часть сервисов, используемая только внутренними потоками работ MDI, была скрыта. В процессе эксплуатации системы обнаружилось, что открытие доступа ко всем сервисам позволяет повысить гибкость и снизить нагрузку на систему. Связано это с тем, что потоки работ в системе основаны на моделях процессов поставщиков и в том случае, когда работа поставщика изменяется, приходится изменять и скрытые процессы.

К сожалению, при начале внедрения системы не удалось получить правдоподобных оценок затрат ресурсов и навыков, необходимых для развёртывания, в ходе интеграции, тестирования и ввода в эксплуатацию. В результате анализа оказалось, что основной источник трудностей – шина сообщений. Для преодоления трудностей потребовалось значительно глубже проникнуть в суть концепций SOA. В конце концов план внедрения системы был соответствующим образом изменён.

При обсуждении рассматриваемого примера среди сотрудников IBM Карл Хэнкок (Carl Hancock), руководитель команды разработчиков сервисов MDI, сделал несколько интересных замечаний, которые полезно было бы рассмотреть наряду с прочими результатами внедрения:

- независимо от того, какая сервисная модель используется, во время интеграции приходится выполнять огромное количество черновой работы, включающей определение процессов, спецификацию транзакций, моделирование данных, а также тестирование, тестирование и ещё раз тестирование. Другими словами, при первых внедрениях следует быть готовым к значительным затратам на выполнение работ;
- корпорация должна найти золотую середину между желанием контролировать процессы поставщика и степенью независимости от этих процессов. Сервис-ориентированная архитектура ничего принципиально не меняет. Она позволяет сделать точки контроля более обозримыми, но, к сожалению, не позволяет устранить переплетение собственных процессов с процессами поставщиков;
- эффективность сервисной модели определяется эффективностью бизнес-процессов, которые реализуются сервисами. Бизнес-процессы без владельцев, чёткого и детального определения (в том числе и метрик) будут «буксовать» и с сервисной моделью, и без неё.

Что же планируется делать дальше? Основная задача – упрощение системы, устранение сложности из описанных ранее интеграционных структур. Во-первых, следует классифицировать поставщиков. Уровень интеграции с использованием MDI подходит не для всех поставщиков, для некоторых из них было бы достаточно использование B2B. Также решено, что использовать несколько MDI для одного поставщика вовсе не обязательно, снижение риска отказа в системе достигается путём увеличения средней наработки на отказ (Mean Time To Failure, MTTF) и использованием для ключевых узлов инфраструктуры

высокой доступности. Как следствие, новая стратегия состоит в использовании одного MDI для каждого поставщика, что значительно упрощает подключение к системе новых узлов.

Для снижения сложности шины сообщений и, как следствие, снижения затрат на тестирование и развёртывание, разработан пилотный вариант ESB, позволяющий получить унифицированный доступ к сервисам с различными интерфейсами. Использование ESB даёт возможность значительно уменьшить объём тестирования, так как шина больше не принимает непосредственного участия в работе существующих сервисов. Также больше не нужно заканчивать все параллельные разработки в один срок, группа разработчиков получает возможность работать более гибко и достигать лучших результатов. Дополнительно к внедрению ESB планируется переход с WebSphere Message Broker на WebSphere Process Server, что позволит осуществить стандартизацию моделей бизнес-процессов. Новые стандартные модели (на BPEL) могут быть напрямую переданы в WPS, что обеспечит более короткое время реакции на изменения в бизнесе.

### **Основные результаты для бизнеса**

Описанный проект продемонстрировал, что SOA может использоваться для разработки ответственных систем и может служить в качестве серьёзной основы для дальнейшего развития. Кроме того, применение SOA повышает процент повторного использования компонентов – внутреннее исследование метрик SOA подтверждает это на протяжении 4-х лет. В подразделении IBM GES в среднем в новых проектах повторно используется до 60% сервисов. Другими словами, в среднем каждые полтора года команда получает нового виртуального разработчика. Большое количество групп разработки ещё сильнее усиливает эффект.

### **Заключение**

Может быть, про рассмотренный проект в какой-то мере можно сказать «Не пытайтесь повторить это дома!». Успех проекта по внедрению виртуального производства – результат обширных навыков и высокого профессионализма команды разработчиков, ориентированной на успех и привыкшей работать над «экстремальными» проектами.

Команда получила дополнительное усиление за счёт осознания бизнесом необходимости внедрения методологии, основанной на SOA, а также за счёт наличия необходимых инструментов и достаточного объёма инвестиций, что позволило достичь поставленных целей в сжатые сроки. Без сомнений, проект мог быть реализован и с помощью более традиционных средств, однако не в такие сроки, не с таким уровнем масштабируемости и не с таким количеством ресурсов. Наконец, проект не мог бы стать успешным без поддержки со стороны руководства. Руководители IBM GES понимали все потенциальные риски, но при этом приняли решение держаться установленного курса. Это очень важное решение – без поддержки руководства и других влиятельных заинтересованных лиц проекты по внедрению SOA обречены на провал.

### ***Система eHub компании Ford***

После успешного использования SOAP-интерфейсов при реализации B2B-приложений, было принято решение об объединении усилий корпоративных архитекторов, проектировщиков производственных систем и команд разработчиков для применения SOAP, XML и других современных стандартов веб-сервисов к решению проблемы взаимодействия информационных систем, как внешних, так и внутренних. В результате был создан eHub – решение для организации взаимодействия на основе стандартов. В eHub используется XML (поверх HTTP), цифровые подписи и OAGIS.

Компания Ford начала использовать eHub для интеграции производственных информационных систем с корпоративными информационными системами и системами, взаимодействующими с поставщиками, в 2001-м году. Первое внедрение в 2001-м году затрагивало две производственные информационные системы, для взаимодействия между

которыми использовались веб-сервисы и событийно управляемый надёжный обмен сообщениями в реальном времени (рис. 26). Среди функций пилотного внедрения было планирование использования материалов, что показало возможность eHub обрабатывать потоки работ, которые затрагивают как системы, так и персонал. Длительность проекта, от начала до ввода системы в эксплуатацию, составила девять месяцев.

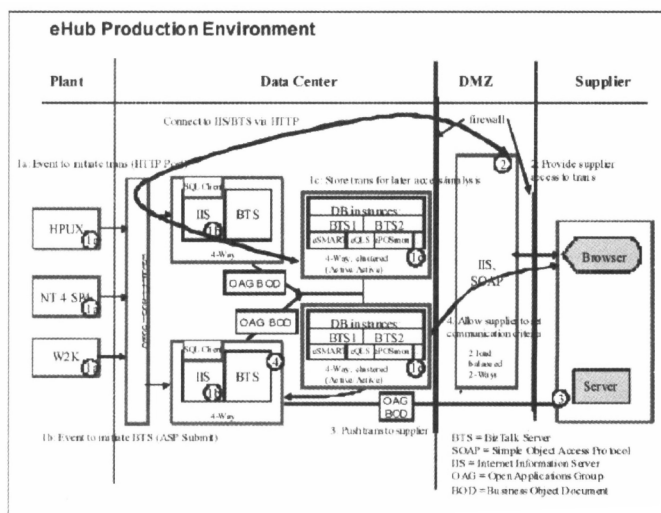
Первоначальный вариант архитектуры был достаточно надёжен, в последующих версиях добавлены дополнительные функции для обеспечения высокой надёжности. Предложенная архитектура дала возможность для ранее недоступных видов взаимодействия между производственными и корпоративными системами.

1. Масштабируемость: данные в формате XML передаются по протоколу HTTP с производств в центр обработки данных. Эффективность передачи XML с помощью HTTP практически совпадает с эффективностью передачи HTML (а в общем случае и превосходит, так как XML не требует передачи встроенной графики). Ford управляет частотой и объёмом XML-посылок. Ford имеет возможность решать, какие данные передавать в реальном времени по событиям, а какие – с получасовым интервалом в пакетах. При необходимости для передачи больших объёмов данных можно использовать FTP. Веб-приложения на сервере B2B обращаются к базе в центре обработки данных, что обеспечивает наилучшие показатели отклика на запросы пользователей.

2. Данные со всех производств и заводов собираются в одном месте, что даёт возможность формирования отчётов о работе нескольких предприятий и формирования запросов «на лету».

В eHub используются несколько документов бизнес-объектов (BOD), определённых стандартом OAGIS. Среди них «Показать использование ресурсов» («Show Consumption», сценарий 60 «Vendor Managed Inventory (Consumption)») и «Показать статус незавершённого производства» («Show WIP Status», сценарий 57 «Production to Manufacturing Execution»).

**Рис. 26. Схема первого внедрения eHub (2001)**



Одно из преимуществ eHub состоит в том, что система была создана с использованием предсказуемого и повторяемого процесса разработки, а также в возможности повторного использования кода для интеграции. Кроме того, благодаря применению открытых стандартных технологий как сама компания Ford, так и её партнёры могут легко и без особых финансовых затрат подключать к eHub свои системы.

Для реализации приложения B2B или P2B разработчики используют инфраструктуру в соответствии со следующим процессом:

- 1) выбор подходящего стандарта XML, или разработка собственного языка, совместимого со стандартами, на основе XML (использование стандартного процесса Ford XML, затрагивающего OAG, STEP и другие основанные на XML стандарты). В состав группы Enterprise Architecture входит небольшая группа архитекторов XML, чья задача заключается в поддержке собственных стандартов XML компании Ford, а также в поддержке разработчиков на этом шаге;
- 2) добавление в приложение возможности формирования сообщений в соответствии с выбранным стандартом. Первоначально данный шаг выполнялся вручную, причём даже в таком случае его выполнение не требовало ощутимых затрат. Позже специалистами Центра компетенции J2EE компании Ford был разработан код на языке Java, который может использоваться для подключения к eHub любого приложения;
- 3) если информация передаётся непосредственно пользователю, следует предоставить ему возможность выбора способа получения: когда и как часто получать сообщения, использовать ли для получения e-mail, пейджер, беспроводное устройство, браузер, специальное приложение, прослушивающее заданный URL и т.д.;
- 4) если в ходе работы требуется взаимодействие с другим приложением, следует использовать стандартный протокол взаимодействия (RosettaNet, OAGIS или более новые стандарты, разработанные Automotive Industry Action Group).

Для поддержки выполнения шагов 3 и 4 в составе eHub имеются обширные средства, поэтому собственные разработки практически не требуются.

Полностью оправдывая своё название, eHub реализует принцип EAI с архитектурой «hub-and-spoke». По этой причине, а также из-за требований поддержки информационных систем, критичных для производства, при проектировании eHub значительное внимание было уделено надёжности. Этот факт во многом предопределил успех проекта, несмотря на некоторые сложности в начале работы.

Вскоре после реализации документа (BOD) «Показать состояние незавершённого производства» для взаимодействия со сторонними поставщиками услуг логистики соответствующий информационный поток был повторно использован для взаимодействия с системой, контролирующей выход автомобилей с завода. В принципе, не было необходимости передавать такую информацию за пределы завода. Однако из-за использования eHub информация передавалась с завода (в Канаде, Мексике или в одном из штатов США) в корпоративный центр обработки данных в Детройте, а затем в обратном направлении. В большинстве случаев глобальная сеть передачи данных оказалась надёжной, но один или два раза возникали сбои связи. В результате появились возражения против использования eHub, по причине которых был разработан проект, предусматривающий замену eHub прямым соединением внутри завода. Однако этот проект так и не был реализован.

Можно сделать вывод, что любая архитектура с концентратором имеет элемент, подверженный риску возникновения отказа, поэтому система с подобной архитектурой будет рассматриваться как источник проблем даже в том случае, если инфраструктура передачи сообщений основным источником возникновения проблем не является (к тому же вероятность отказа концентратора может быть существенно снижена путём использования средств обеспечения высокой доступности).

В 2002-м году группой Enterprise Architecture выполнено развёртывание обновленной версии eHub, обладающей возможностью подключения большего числа систем. Производственные приложения были переведены на использование новой версии, а серверы, применявшиеся для первого внедрения – перепрофилированы. В 2003-м году, спустя всего год после первого развёртывания, eHub обрабатывал пять миллионов сообщений в месяц с



возможностью расширения этого количества. eHub использовался в 20 проектах для решения задач управления кадрами, маркетинга, разработки продукции, управления поставками и производства. С его помощью компания Ford оказалась более тесно интегрирована с поставщиками компонентов и логистических услуг, а также с дилерами. eHub успешно используется в Северной Америке, Европе и Азии.

Область использования eHub продолжает расширяться. В 2006-м году eHub использовался как минимум на 24 заводах в Северной Америке и Европе. Девять компаний-перевозчиков получали в реальном времени информацию о количестве выпущенных автомобилей с 18 сборочных производств в Канаде, Мексике и США. eHub – всего лишь один пример вклада компании Ford в совершенствование производства и инновации.

## Приложение В. Авторы и рецензенты руководства

### Авторы

**Alan Boyd**

IBM Corporation  
561-862-2774  
[alboyd@us.ibm.com](mailto:alboyd@us.ibm.com)

**David Noller**

IBM Corporation  
866-405-7060  
[nollerd@us.ibm.com](mailto:nollerd@us.ibm.com)

**Paul Peters**

IBM Corporation  
877-760-8247  
[pdpeter@us.ibm.com](mailto:pdpeter@us.ibm.com)

**Dave Salkeld**

IBM Corporation  
919-882-6110  
[salkeld@us.ibm.com](mailto:salkeld@us.ibm.com)

**Tim Thomasma**

Capgemini  
734-730-9112  
[Tim.Thomasma@capgemini.com](mailto:Tim.Thomasma@capgemini.com)

**Charlie Gifford**

21st Century Manufacturing Solutions LLC  
208-788-5434  
[charlie.gifford@cox.net](mailto:charlie.gifford@cox.net)

**Steven Pike**

IBM Corporation  
802-769-3424  
[srpikes@us.ibm.com](mailto:srpikes@us.ibm.com)

**Alison Smith**

AMR Research, Inc.  
617-574-5213  
[ASmith@amrresearch.com](mailto:ASmith@amrresearch.com)

### Рецензенты

**George Hudson**

IBM

**Bas Pluim**

IBM

**Aaron LaBella**

IBM

**Julie Fraser**

Industry Directions Inc.

**Paul Ashmore**

Independent MES Consultant

**David R Hinkler**

Rockwell Automation

**Mohammed Zuhair**

Rockwell Automation

**Alicia Bowers**

GE Fanuc

**Kay Freund**

IBM

**Victor Valle**

IBM

### Русский перевод

**А.П. Козлецов**

Российская рабочая группа MESA  
International  
[A.Kozletsov@mesarussia.ru](mailto:A.Kozletsov@mesarussia.ru)

## Приложение С. Список сокращений/гlossарий

A2A – Application to Application (взаимодействие двух информационных систем)

AIAG – Automotive Industry Action Group (Группа стандартизации в автомобильной промышленности)

APS – Advanced Planning and Scheduling (система детального планирования и составления расписаний)

B2B – Business to Business (взаимодействие между фирмами или корпоративными информационными системами)

B2MML – Business to Manufacturing Markup Language (язык разметки для взаимодействия между корпоративными и производственными информационными системами)

BatchML – Batch Markup Language (язык разметки для описания рецептурных (Batch) производств)

BI – Business Intelligence (бизнес-аналитика)

BOD – Business Object Definition (описание бизнес-объекта для использования в обмене информацией)

BPML – Business Process Execution Language (язык исполнения бизнес-процессов)

BPM – Business Process Management (управление бизнес-процессами)

CAD – Computer Aided Design (автоматизированное проектирование)

CAE – Computer Aided Engineering (автоматизированная разработка)

CAM – Computer Aided Manufacturing (автоматизированное производство)

Canonical (канонический) – соответствующий стандартам и принятым правилам, способный отобразить значение в простой форме

CAPA – Corrective Action and Preventive Action (корректирующие и предупреждающие действия)

CBE – Common Based Event – стандарт сообщений для управления распределёнными веб-сервисами (WSDM) - [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wsdm](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsdm)

CID – Change in Design (изменения в процессе проектирования)

CMM – Coordinate Measuring Machine (координатно-измерительная машина)

CMMS – Computer Maintenance Management System (система управления обслуживанием ЭВМ)

CORBA – Common Object Request Broker Architecture (обобщённая архитектура брокера объектных запросов)

COTS – Commercial off-the-shelf («коробочное» программное обеспечение)

CRM – Customer Relationship Management (управление взаимодействием с потребителем)

DCS – Distributed Control System (распределённая система управления)

DTD – Document Type Definition (определение типа документа)

DMZ – Demilitarized Zone (демилитаризованная зона – часть сети за сетевым экраном, содержащая компьютеры с IP-адресами, доступными из Интернет)

ECE – Enterprise Composition Environment (среда для создания композитных систем предприятия)

ECO – Engineering Change Order (запрос на изменение проекта)

eHub – название концентратора в примере интеграции от компании Ford

EDMS – Enterprise Document Management System (корпоративная система управления документооборотом)

EDW – Enterprise Data Warehouse (корпоративное хранилище данных)

EIMS – Enterprise Integration Messaging Specification (корпоративная спецификация формата обмена сообщениями)

EJB – Enterprise JavaBeans (компоненты для создания серверных приложения на Java)

enum – перечислимый пользовательский тип данных в языках программирования наподобие C++

ERP – Enterprise Resource Planning (управление планированием ресурсов предприятия)  
ESB – Enterprise Service Bus (общая сервисная шина предприятия)  
EWI – Electronic Work Instructions (электронные инструкции)  
FTP – File Transfer Protocol (протокол передачи файлов)  
GUI – Graphical User Interface (графический интерфейс пользователя)  
IEC – International Engineering Consortium (Международный консорциум инженеров)  
IIOP – Internet Inter-ORB Protocol (протокол передачи сообщений между объектами по TCP/IP)  
IMS – Integration Messaging Specification (спецификация обмена сообщениями для интеграции)  
ISA – Instrumentation, Systems and Automation Society (Общество измерений, систем и автоматизации – организация, выпускающая стандарты в области управления производственными операциями и автоматизации. В последние годы именуется International Society for Automation – международное общество автоматизации)  
ISA-88 – стандарт описания рецептурного (batch) производства  
ISA-95 Enterprise – спецификация интеграции систем управления ISA-dS95.01-1999  
J2EE – Java 2 Enterprise Edition  
JDBC – Java Database Connectivity – технология работы с базами данных программ, написанных на Java  
JMS – Java Message Service (служба передачи сообщений для Java)  
LIMS – Laboratory Information Management System (система управления лабораторной информацией)  
MCE – Manufacturing Composition Environment (среда для создания композитных систем производства)  
MDI – Multi-Data Source Integrator (интегратор данных из различных источников – название концентратора в примере IBM)  
MDM – Master Data Management (управление нормативно-справочной информацией)  
Mfg.MDM – Manufacturing Master Data Management (управление производственной нормативно-справочной информацией)  
Mfg.Ops – Manufacturing Operations (производственные операции)  
MES – Manufacturing Execution System (система оперативного управления производством)  
MIMOSA – Machine Information Management Open Systems Alliance (сообщество открытых систем управления машинной информацией)  
MOM – Manufacturing Operations Management (управление производственными операциями – приложения уровня 3 модели ISA-95)  
MSB – Manufacturing Service Bus (производственная сервисная шина, ESB, расширенная средствами передачи производственных данных – например, специфическими сервисами и средствами поддержки стандартов интеграции)  
MTTF – Mean Time To Failure (средняя наработка на отказ)  
NPI – New Product Introduction (представление новой продукции)  
OAG – Open Applications Group (группа открытых приложений)  
OAGi – Open Applications Group, inc (группа открытых приложений)  
OAGIS – Open Applications Group Integration Specification (спецификация интеграции группы открытых приложений)  
OI – Operations Intelligence (операционная аналитика)  
OPC – Open Linking and Embedding (OLE) for Process Control (OLE для управления процессом)  
OPM – Operations Process Management (управление процессом выполнения операций)  
P2B – Plant to Business (связь корпоративных и производственных систем)  
P&PLM – Product Process Lifecycle Management (управление жизненным производственного процесса)

PLC – Programmable Logic Controller (программируемый логический контроллер)  
PLM – Product Lifecycle Management (управление жизненным циклом продукции)  
QOS – Quality of Service (качество обслуживания)  
RFID – Radio Frequency Identification (идентификация по радиочастоте)  
RPC – Remote Procedure Call (удалённый вызов процедур)  
SCA – Service Component Architecture (Сервисная компонентная архитектура)  
SCADA – Supervisory Control and Data Acquisition (система диспетчерского управления и сбора данных)  
SCM – Supply Chain Management (управление цепочками поставок)  
SDO – Service Data Object (объект хранения данных сервисов)  
SOAP – Simple Object Access Protocol (упрощённый протокол доступа к объектам)  
SPC – Statistical Process Control (статистическое управление процессом)  
SSL – Secure Socket Layer (уровень безопасных сокетов)  
TBG – Trade and Process Business Group (группа торгового и производственного бизнеса)  
TCIOps – Tool Control Infrastructure Operations (инфраструктурные операции управления инструментом)  
TLS – Transport Layer Security (безопасность транспортного уровня)  
TREAD – Transportation Recall Enhancement, Accountability and Documentation (акт федерального правительства США)  
UDDI – Universal Description, Discovery and Integration (универсальное описание, поиск и обнаружение)  
UI – User Interface (пользовательский интерфейс)  
UN/CEFACT – United Nations Centre for Trade Facilitation and Electronic Business (Комитет ООН по содействию торговле и электронному бизнесу)  
UPS – Uninterruptable Power Supply (источник бесперебойного питания)  
VMF – Virtual Manufacturing Framework (каркас виртуального производства)  
VMI – Vendor-Managed Inventory (склад, управляемый поставщиком)  
VoIP – Voice over Internet Protocol (передача голосовых данных посредством протокола IP)  
W3C – World Wide Web Consortium (консорциум World Wide Web)  
WBF – World Batch Forum (всемирный форум специалистов рецептурного производства)  
WCF – Windows Communication Foundation (средство организации обмена сообщениями в системах Windows)  
WS-BPEL – Web Services Business Process Execution Language (язык исполнения бизнес-процессов для веб-сервисов)  
WSDL – Web Services Description Language (язык описания веб-сервисов)  
WSDM – Web Services Distributed Management (распределённое управление веб-сервисами)  
WS-I – Web Services Interoperability (взаимодействие веб-сервисов)  
WSRR – реестр и репозиторий сервисов WebSphere  
XML – eXtensible Markup Language (расширяемый язык разметки)  
XPath – XML Path Language (язык запросов к элементам XML-документа)  
XSD – XML Schema Definition (описание схемы XML)  
XSLT – Extensible Stylesheet Language Translation (расширяемый язык преобразования таблиц стилей)

## **Приложение D. Товарные знаки**

DB2 – зарегистрированный товарный знак корпорации IBM  
IEC – зарегистрированный товарный знак Международного консорциума инженеров  
ISA – зарегистрированный товарный знак Общества измерений, систем и автоматизации  
Microsoft – зарегистрированный товарный знак корпорации Microsoft  
MQSeries – зарегистрированный товарный знак корпорации IBM

OAGIS – зарегистрированный товарный знак Open Applications Group, Inc  
OPC – зарегистрированный товарный знак OPC Foundation  
RosettaNet – зарегистрированный товарный знак консорциума RosettaNet  
Tivoli – зарегистрированный товарный знак корпорации IBM  
WebSphere – зарегистрированный товарный знак корпорации IBM  
WebSphere Process Server – зарегистрированный товарный знак корпорации IBM  
Windows – зарегистрированный товарный знак корпорации Microsoft

## **Приложение Е. Список источников**

- 1 OAGi White Paper – Plug and Play Business Software Integration, The Compelling Value of the Open Applications Group, David Connelly, CEO, Open Applications Group, Inc, 1995.
- 2 The Open Applications Group Integration Specification, Michael Rowell, Open Applications Group, Inc, June, 2003.
- 3 XML Excellence at Ford Motor Company, Tim Thomasma, XML Journal, Volume 3, Issue 9, September, 2002.
- 4 OAGIS 8: Practical integration meets XML Schema, Mark Feblowitz, XML Journal, Volume 3, Issue 9, September, 2002.
- 5 XML in the Auto Industry: Summer 2002, Pat Snack and Sig Handleman, XML Journal, Volume 3, Issue 9, September, 2002.
- 6 ANSI / ISA-88.01, Batch Control Part 1: Models and Terminology, ISA, 2000.
- 7 ANSI / ISA-88.00.02, Batch control - Part 2: Data Structures and Guidelines for Languages, ISA 2001.
- 8 ANSI / ISA-95.00.01, Enterprise-Control System Integration Part 1: Models and Terminology, ISA 2000.
- 9 ANSI / ISA 95 00.02, Enterprise-Control System Integration Part 2: Object Model Attributes, ISA 2000.
- 10 ISA-88, Batch Control- Part 3: General and Site Recipe Models and Representation, ISA 2002.
- 11 ANSI / ISA-95.00.03, Enterprise-Control System Integration Part 3: Models of Manufacturing Operations Management, 2005.
- 12 ISA-88, Batch Control - Part 4: Batch Production Records, ISA 2006.
- 13 ISA-95 Part 5 (Draft 10), Business-to-Manufacturing (B2M) Transactions, ISA 2006.
- 14 WebSphere Service Registry and Repository Handbook, ISBN: 0738489972
- 15 Standards for Manufacturing Systems Integration, ISA-95 and OAGIS White Paper Series, White Paper #2: OAGIS, ISA-95 and Related Manufacturing Integration Standards, A Survey, ISA/MESA Publication, 2/1/2007
- 16 Standards for Manufacturing Systems Integration, ISA-95 and OAGIS White Paper Series, White Paper #1: An Overview and Comparison, ISA/MESA Publication, 2/1/2007

## Приложение F. Частичный список соответствия типов данных J2EE и .NET

Simple Type	Java Type	.NET Type
xsd:string	java.lang.String	String
xsd:integer	java.math.BigInteger	Int64 ?
xsd:int	Int	Int32
xsd:long	Long	Int64
xsd:short	Short	Int16
xsd:decimal	java.math.BigDecimal	Decimal
xsd:float	float	Single
xsd:double	double	Double
xsd:boolean	boolean	Boolean
xsd:byte	byte	SByte
xsd:QName	javax.xml.namespace.QName	String ?
xsd:dateTime	java.util.Calendar	DateTime
xsd:base64Binary	byte[]	Byte(Array)
xsd:hexBinary	byte[]	Byte(Array)
xsd:time	java.util.Calendar	DateTime ?
anyURI	java.lang.String	System.Uri ?
anySimpleType	java.lang.String	String ?
xsd:date	java.util.Calendar	DateTime ?
xsd:negativeInteger	java.math.BigInteger	System.Decimal
xsd:nonNegativeInteger	java.math.BigInteger	System.Decimal
xsd:nonPositiveInteger	java.math.BigInteger	System.Decimal
xsd:unsignedInt		UInt32
xsd:positiveInteger	java.math.BigInteger	System.Decimal ?
xsd:unsignedLong	java.math.BigInteger	UInt32/UInt64 ?



IBM – крупнейшая мировая компания, занимающаяся информационными технологиями. На протяжении своей 80-летней истории IBM является лидером в предоставлении услуг для инноваций в бизнесе. Используя собственный опыт, а также опыт ключевых партнёров IBM предлагает широкий спектр услуг, решений и технологий различной сложности, позволяющих на полную мощь использовать возможности электронного бизнеса.



Capgemini – один из мировых лидеров в оказании консультационных услуг, разработке технологий и аутсорсинге. Capgemini помогает клиентам наилучшим образом использовать имеющиеся технологии. Capgemini даёт клиентам идеи и возможности, помогающие достичь впечатляющих результатов путём использования уникального способа организации работ – Collaborative Business Experience, а также глобальной модели поставок под названием Rightstore, позволяющей поставлять нужные ресурсы в нужное место по доступной цене. Компания присутствует в 36 странах. В 2007 году компания имела оборот 8,7 миллиарда евро (примерно 12 миллиардов долларов) и обеспечивала работой 83000 человек по всему миру.



MESA International (Ассоциация MES) способствует обмену информацией об успешных внедрениях, стратегиях и инновациях в управлении производственными процессами, управлении качеством и оптимизации производства. Мероприятия, проводимые MESA, а также публикации, помогают производителям, системным интеграторам и поставщикам решений достичь новых высот в реальных внедрениях, комбинирующих бизнес-процессы, информационные процессы, производственные процессы, цепочки поставок и т.д. Более подробную информацию можно получить на сайте: [www.mesa.org](http://www.mesa.org).





# Как ошибки оценки в ERP помогают распространению MES

Применение модели MESA делает возможной интеграцию ERP и MES для повышения эффективности производства

Карл Шнеебауэр, вице-председатель MESA EMEA



## СОДЕРЖАНИЕ

Введение .....	63
1. Расчёт по центрам затрат и ошибки, связанные с фиксированной почасовой ставкой .....	63
2. Расчёт по единицам калькуляции (расчёт себестоимости) и потребность в агрегированных данных ..	64
3. Регистрация фактического потребления материалов по ретроспективным данным: устаревший подход.....	67

## ВВЕДЕНИЕ

Зачастую руководители предприятий пытаются улучшить производственные процессы и цепочку создания добавленной стоимости на основе только тех данных, которые содержатся в ERP-системах. При этом они сталкиваются с тремя серьёзными проблемами:

- при определении стоимости по центрам затрат используются неточные и произвольные оценки времени работы оборудования;
- для работы методов расчёта стоимости по единицам калькуляции необходимы точные данные о времени исполнения, остатках и браке по каждому заказу;
- регистрация материалов, не потребленных фактически при выполнении заказа, приводит к необходимости постоянных инвентаризаций.

Подобные проблемы характерны для всех ERP-систем, поэтому их наличие создаёт большой рынок для систем управления производственным процессом – MES-систем.

Некоторые разработчики ERP-систем осознают проблему, связанную с недостаточным использованием данных о работе цехов для оптимизации производственных процессов. Первой стала компания SAP, в 2008 году приобретшая разработчика MES Visiprise. Теперь это программное обеспечение носит название SAP ME и применяется только для обработки данных с производств. Обмен информацией с SAP ERP производится через специальные тонкие интерфейсы. Остальные разработчики ERP последуют примеру SAP, так как их клиенты с производственных предприятий требуют наличия возможности автоматического сбора данных с оборудования, оперативного управления работами, выполнения сложных процедур статистического управления процессам, а также возможности отслеживания продукции и материалов. Такие задачи с трудом решаются при использовании ручного ввода данных в ERP, поэтому процедуры, использующие ручной ввод, вскоре перестанут применяться на производственных предприятиях.

Решение главной задачи ERP – оптимизации финансовых потоков – чрезвычайно затрудняется из-за произвольных предположений о величине основных показателей и ошибочных данных, появившихся в результате ручного ввода информации. Поэтому естественно задать вопрос: «Зависит ли ERP от MES в плане точности данных?». Мой ответ – «Да!». Для того чтобы его аргументировать, рассмотрим три системные проблемы более подробно.

## 1. РАСЧЁТ ПО ЦЕНТРАМ ЗАТРАТ И ОШИБКИ, СВЯЗАННЫЕ С ФИКСИРОВАННОЙ ПОЧАСОВОЙ СТАВКОЙ

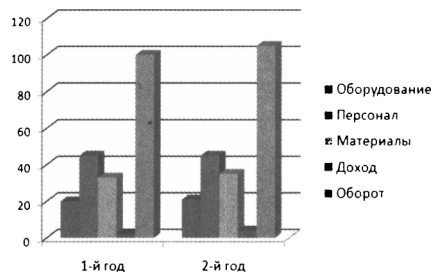
При расчёте по центрам затрат производственное предприятие рассматривается как совокупность участков. Предположим, что имеется шесть станков с ЧПУ, двадцать штамповочных машин, шесть сварочных роботов и двенадцать сборочных рабочих мест. Каждый участок рассматривается как центр затрат, состоящих из стоимости эксплуатации оборудования или рабочих мест, зарплаты персонала и дополнительных затрат, связанных, например, с инструментами, применяемыми только на этом участке. Затем, в соответствии со сменной организацией работ для каждого центра затрат вычисляется почасовая ставка, имеющая постоянную величину. На основе этой ставки определяются доходы и затраты каждого нового заказа.

С точки зрения MES, применение фиксированной почасовой ставки (например, \$86 за час) ошибочно по определению. Связано это с тем, что благодаря MES 10-15% дополнительных заказов может быть выполнено теми же рабочими на том же оборудовании, вследствие чего оценка значения почасовой ставки снизится на 10-15%. Другая проблема

состоит в том, что никак не учитываются периоды отсутствия заказов. При работе в три смены станок с ЧПУ работает 130 часов, на основании чего можно заключить, что в год он работает 6500 часов. Однако каково реальное годовое время работы станка? Если оно составит хотя бы 4000 часов, то для достижения той же самой точки безубыточности почасовая ставка должна быть поднята до \$100. Подобные эффекты дают широкий простор для повышения рентабельности. Например, субботняя смена, организованная в соответствии с концепциями бережливого производства, в ходе которой выполняется чуть больше заказов, а наблюдение за оборудованием производят лишь несколько рабочих, легко может увеличить ежегодное время работы оборудования до 7000 часов. Почасовая ставка в этом случае составит \$75. Такой подход широко известен среди производителей, однако его эффект нивелируется расчётом по центрам затрат в ERP-системах.

Рассмотрим пример на рис. 1. Компания может увеличить свой доход на 100% путём увеличения объёма выполняемых заказов всего на 5%. Такая ситуация характерна для большинства производственных предприятий среднего размера, поэтому очень важно иметь возможность ответить на такой вопрос: «Как мы можем выполнять больший объём работы, как мы можем снизить объём брака и увеличить оборот?».

**Рис.1.**  
**Повышение**  
**дохода на**  
**100% при**  
**увеличении**  
**числа заказов**  
**на 5%**



	1-й год	2-й год
Оборудование	20	21
Персонал	45	45
Материалы	33	35
Доход	2	4
Оборот	100	105

График на рис. 1 вполне обычен, так как большинство производственных предприятий работает с невысокой прибылью. Поэтому эффект от внедрения функций, определённых в модели MESA – таких как автоматический мониторинг оборудования и оперативное планирование – может быть очень серьёзным.

Все оценки по центрам затрат, сделанные на уровне ERP, должны постоянно корректироваться в соответствии с информацией, полученной MES в ходе очередного производственного цикла. Необходимо не просто передавать данные с нижнего уровня на верхний, а обеспечить циклический обмен информацией между MES и ERP, как показано на рис. 2. Только так можно обеспечить корректную оценку по центрам затрат.

## 2. РАСЧЁТ ПО ЕДИНИЦАМ КАЛЬКУЛЯЦИИ (РАСЧЁТ СЕБЕСТОИМОСТИ) И ПОТРЕБНОСТЬ В АГРЕГИРОВАННЫХ ДАННЫХ

Второй подход, применяемый в ERP, основан на учёте как стоимости самого продукта, так и операций, выполняемых при производстве.

Без использования функций MES корректно выполнить расчёт себестоимости практически невозможно, так как основная задача состоит в том, чтобы определить для каждого этапа производственного процесса следующие величины:

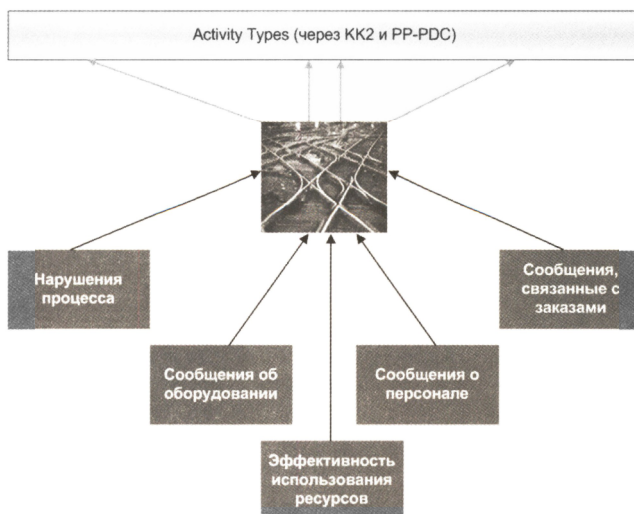
**Рис.2.**  
**Модель**  
**MESA**



- трудозатраты;
- время работы оборудования;
- уровень брака;
- уровень вторичной переработки;
- стоимость используемых инструментов;
- и т.д.

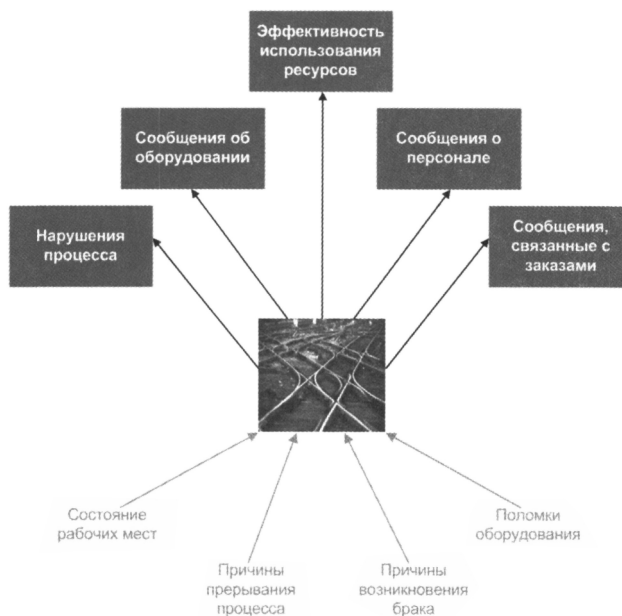
На рис.3 концепция расчёта себестоимости показана на примере типов деятельности (Activity Types) SAP. Для того чтобы обеспечить эффективный расчёт себестоимости на уровне ERP, информация о заказах, времени работы оборудования, поломках оборудования, персонале, нарушениях производственного процесса, величине показателей, определяющих количество отходов на один заказ (недостаток инструментов, вторичная переработка, ремонт в процессе работы...) должна поступать в агрегированном виде.

**Рис. 3. Данные для**  
**расчёта себестоимости**



Поэтому ниже уровня ERP должен располагаться MES-уровень, на котором производится обработка данных о ходе производственного процесса.

**Рис. 4. Агрегация данных с MES-уровня**



В настоящее время наблюдается интенсивный переход от работы на склад к способам организации производства, более ориентированным на потребности конкретного потребителя, индивидуализации упаковки продукции, концепции «Производство после оплаты» («Pay First Produce Then»). При этом сложности расчёта себестоимости в ERP становятся непреодолимыми. В результате расчёты приходится производить практически вручную – при помощи ручки и бумаги или электронных таблиц.

Заслуживает одобрения тот факт, что в SAP осознали эту тенденцию и приобрели Visiprise, ставшую теперь SAP ME. Очевидно, что у концепции расчёта себестоимости на основе данных, введённых вручную с помощью клиента SAP GUI, нет будущего. Громадный объём данных в системах управления технологическими процессами, непостоянные и многогранные требования к распределению заказов и вторичной обработке вынудили SAP принять стратегическое решение о переносе обработки данных о ведомостях материалов и расписаниях на уровень SAP ME – MES-уровень.

Компания SAP стала первой, однако другие поставщики ERP-систем, несомненно, последуют её примеру, поскольку всё увеличивающиеся требования к точности расчёта себестоимости невозможно удовлетворить с помощью ручного ввода данных. Так что другим крупным игрокам на поле ERP придётся вслед за SAP перемещаться ближе к MES для того, чтобы обеспечить свои продукты корректными данными.

### 3. РЕГИСТРАЦИЯ ФАКТИЧЕСКОГО ПОТРЕБЛЕНИЯ МАТЕРИАЛОВ ПО РЕТРОСПЕКТИВНЫМ ДАННЫМ: УСТАРЕВШИЙ ПОДХОД

Большинство производителей страдает от постоянных инвентаризаций, проводимых вручную, уровни запасов в ERP-системах постоянно отклоняются от фактических. Причина этого в том, что реальное количество материалов, использованных при производстве продукции, не регистрируется теми, кто продукцию производит.

Консультанты по ERP советуют использовать достаточно понятный, но логически небезупречный принцип обратного прилива, в соответствии с которым «наиболее вероятное» количество материалов, требуемых для выполнения одного заказа, автоматически вычитается из текущего уровня запасов. Так делать довольно рискованно в силу того, что результат сильно зависит от корректности нормативно-справочной информации. Также он зависит от того, насколько правдивы данные о количестве отходов, а такие данные вводятся вручную. Как правило, все последующие оценки строятся на количестве произведённой продукции и величине отходов. Пусть, например, для производства каждого из 1000 шоколадных батончиков в соответствии с ведомостью материалов требуется 30,14 граммов какао-крема. Если оператор указал, что произведено 1000 батончиков и ещё 40 батончиков оказались бракованы, то в соответствии с принципом обратного прилива будет зарегистрировано потребление 1040х30,14 грамм, то есть 31,35 килограмма какао-крема. Однако возможно, что оператор слил 23 кг в процессе наладки оборудования. Могла возникнуть потребность в замене сломавшегося устройства для формования батончиков, что привело бы к потере ещё 38 кг. Оставшиеся 112 кг какао-крема могли быть использованы для выполнения следующего заказа. Как принцип обратного прилива может учесть всё это?

Вдобавок, в цехах часто имеются небольшие, но неучтённые собственные запасы материалов, которые никак не отражены в информационных системах. Работники учатся обходить систему учёта, в результате чего последняя уже не может предоставлять корректные данные. Для того, чтобы исправить ситуацию необходимо проводить ручную инвентаризацию.

Помочь здесь могут лишь решения, включающие получение данных напрямую от весов или производственного оборудования. Постоянно уменьшающиеся размеры отдельных заказов и повышающийся уровень автоматизации производства требуют точного отслеживания потребления материалов.

С учётом того, что большинство ERP-систем до сих пор работают по устаревшему принципу обратного прилива, важность MES-уровня, поставляющего точные и реальные данные, становится очевидной.

В заключение ещё раз следует сказать о том, что только MES-уровень способен обеспечить решение трёх системных проблем. Только Стратегические Инициативы модели MESA могут помочь руководителям производственных предприятий решить эту головоломку, применив комплексный подход.

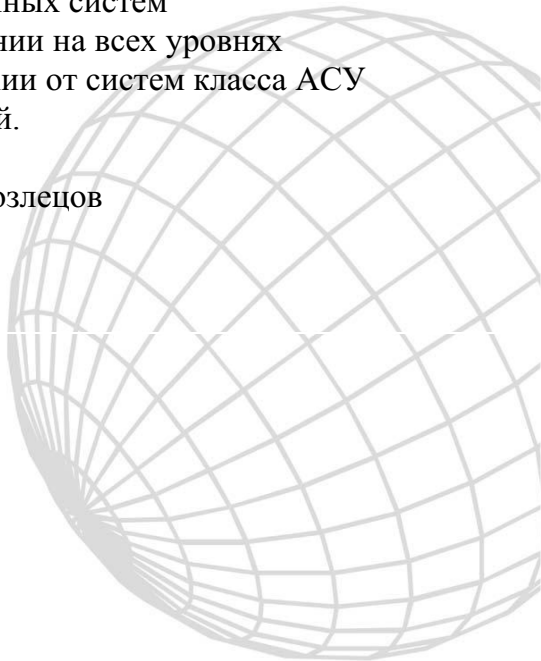
Карл Шнеебауер (Karl Schneebauer), управляющий партнер компании MPDV Mikrolab GmbH, вице-президент MESA Europe. Международно-признанный эксперт и известный докладчик по эффективности производства и непрерывным улучшениям. Автор многочисленных публикаций по этим тематикам. Руководит несколькими международными проектами по глобальному развертыванию MES-систем на предприятиях крупных международных компаний.  
K.Schneebauer@mpdv.de



# Стандарты и технологии интеграции производственных информационных систем

В статье рассматриваются стандартные подходы к интеграции информационных систем производственной компании на всех уровнях производственной иерархии от систем класса АСУ ТП до бизнес-приложений.

И.С. Решетников, А.П. Козлецов





## СОДЕРЖАНИЕ

Введение .....	5
Интеграция систем управления технологическим процессом .....	6
Интеграция с MES-уровнем.....	6
Интеграция бизнес-приложений .....	6
Обмен документами .....	6
Интеграция на уровне модели данных .....	6
Интеграция на основе единой НСИ.....	6
Сервис-ориентированный подход.....	6
Обмен сообщениями .....	6
Интеграция с внешними системами .....	6
Выводы .....	6
Список литературы.....	6

## ВВЕДЕНИЕ

Сегодня ни одно современное предприятие невозможно представить без множества информационных систем, выполняющих самые разнообразные задачи - от выдачи команд на исполнительные устройства до прогнозирования перспективного развития корпорации на несколько лет. Во всех этих системах обрабатывается и собирается множество самых разнообразных данных, формируется множество документов и появляется информация, которая является входной для работы других систем. Во все времена идея объединить существующие системы в единый комплекс была актуальна, и сегодня словосочетание «интеграция разнородных информационных систем» или что-то созвучное есть в арсенале практически любого системного интегратора. Однако, основная проблема кроется в том, что понятие «интеграция» не имеет однозначного определения, и Вам могут предложить что угодно - от систем уровня SAP XI до обменных текстовых файлов в формате csv, - и всё это действительно будет относиться к интеграции.

В то же время, во всём мире уже выработан целый ряд стандартных, подчёркиваем, стандартных подходов к объединению нескольких систем в единый комплекс. Знание и понимание различных стандартов и области их применения может существенно помочь при выборе подходов по построению действительно интегрированной информационной среды. Сразу оговоримся, что в данной работе мы не касаемся таких вопросов, как целесообразность интеграции, оценка её эффективности и пр., как например, это декларируется стандартом ISO 27001, а говорим исключительно о её технических аспектах. Объектом обсуждения будет типовая производственная компания, имеющая все уровни производственной иерархии - от исполнительных механизмов под управлением ПЛК (Программируемый Логический Контроллер) до систем бизнес-аналитики и средств совместной (кооперативной) работы с другими предприятиями.

На современном производственном предприятии используются информационные системы самых разных производителей. Задачи, которые решаются этими системами, также различны, но зачастую оказываются взаимосвязанными, в результате чего возникает необходимость обмена данными между разными системами. Несмотря на то, что нельзя указать единый подход, руководствуясь которым можно интегрировать две произвольные системы, существует большое количество стандартов и технологий, которые могут значительно облегчить решение задачи интеграции [1]. Им и посвящена данная работа. Для наглядности все уровни интегрируемых систем, описываемые в настоящей статье, собраны на одном рисунке (рис. 1), статья фактически является лишь комментарием к этой схеме.

## ИНТЕГРАЦИЯ СИСТЕМ УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ ПРОЦЕССОМ

Как видно из рис. 1 необходимость решения задач интеграции возникает уже на самом нижнем уровне иерархии управления. Здесь нужно обеспечить возможность обмена информацией между ПЛК и различными датчиками и исполнительными устройствами, а также между различными ПЛК. В зависимости от объёма данных, которые должны передаваться от датчика к контроллеру и от контроллера к исполнительному устройству, а также от «интеллектуальности» таких устройств, для обеспечения обмена информацией могут использоваться полевые (HART, ASi, PROFIBUS PA, Foundation Fieldbus, DeviceNet, CAN Open) или промышленные (Modbus, PROFIBUS DP, PROFINET, DH-485, DF1) сети. К полевым сетям с некоторыми оговорками можно отнести и специальные информационные сети, используемые при автоматизации зданий - такие как LonWorks или KNX/Instabus [2].

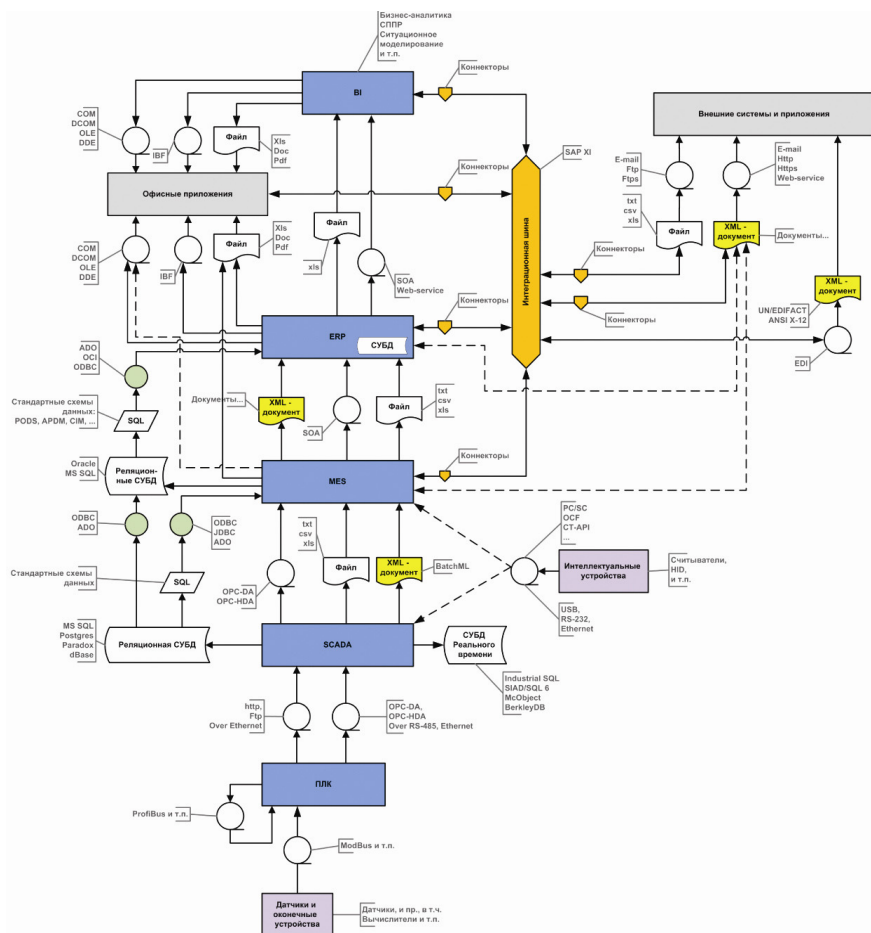


Рис. 1. Уровни интеграционного взаимодействия

Полевые сети предназначены в первую очередь для организации связи между контроллерами и исполнительными устройствами, либо между контроллерами и удалёнными модулями ввода-вывода, в то время как основная задача промышленных сетей - обеспечить связь контроллеров между собой и с рабочими станциями операторов [3]. Несмотря на разграничение задач между двумя типами сетей, провести чёткую границу между ними возможно далеко не всегда. Например, PROFIBUS DP широко применяется как для межконтроллерного взаимодействия и связи с системами верхнего уровня, так и для связи контроллера с распределённой периферией и интеллектуальными полевыми приборами (частотными приводами, локальными регуляторами и пр.). То же самое можно сказать и про Modbus. В большинстве случаев эта технология применяется для опроса и управления полевыми приборами, однако нередко приходится сталкиваться и с примерами межконтроллерной связи по Modbus [4].

Полевые сети, как правило, обладают довольно простой структурой и имеют относительно низкую стоимость монтажа и наладки. Кроме того, обычно только полевые сети имеют характеристики, позволяющие использовать их во взрывоопасных зонах. Применяются полевые сети не только для замены аналоговых линий связи, но и для расширения функциональности таких линий (передача диагностических сообщений, настройка прибора). Скорость передачи данных в полевых сетях обычно не превышает 9600 бит/с. Промышленные же информационные сети могут иметь сложную топологию, объединять большое количество абонентов и позволяют обмениваться информацией на очень больших скоростях - 12 мбит/с (PROFIBUS DP), 100 мбит/с (PROFINET) [5, 6].

Но контроллер должен общаться не только с полевыми приборами, но и с контроллерами, управляющими работой других участков производственной линии. Задача интеграции отдельных ПЛК легко решается только в том случае, если объединяются ПЛК одного производителя. В противном случае возникают трудности даже в том случае, если ПЛК поддерживают одни и те же технологии связи, например, PROFIBUS DP. В течение долгого времени большой популярностью пользовалась организация связи между ПЛК с помощью дискретных и аналоговых сигналов. То есть выходы одного контроллера соединялись со входами другого, полученные линии связи использовались для передачи сигналов о готовности участка производственной линии к обработке заготовок, команд в дискретном коде и т.д. Недостатки данного способа очевидны: низкая гибкость, ограниченный объем передаваемой информации, малая масштабируемость. Но из-за главного достоинства - возможности обеспечить связь между контроллерами любых производителей - данный способ применяется и сейчас.

Современный подход заключается либо в использовании для связи ПЛК промышленных информационных сетей, либо в организации обмена информацией через системы верхнего уровня - например, SCADA (Supervisory Control And Data Acquisition - Диспетчерский контроль и сбор данных) [7]. Промышленная информационная сеть должна обладать или возможностью обмена данными между одноранговыми устройствами, или возможностью одному и тому же контроллеру работать и в качестве ведущего устройства при обмене данными с полевыми устройствами, и в качестве ведомого - при обмене данными с другими контроллерами. Эти возможности обеспечивают, среди прочих, сети PROFIBUS DP, PROFINET, DF1 [8]. Как говорилось выше, на практике так объединяют в основном контроллеры одного производителя.

При обмене данными между ПЛК, как и между системами более высокого уровня, важно не только обеспечить передачу от одного участника обмена к другому, но и добиться того, чтобы одни и те же данные интерпретировались всеми участниками одинаково. Эта задача становится особенно важной в том случае, когда необходимо объединить между собой автоматизированные производственные линии разных производителей. Даже если на линиях используются одинаковые ПЛК, интеграция может быть затруднена, например, из-за различных форматов передаваемых данных. Одно из решений этой проблемы - использование концепции компонентной автоматизации (Component Based Automation, CBA) [8]. В соответствии с этой концепцией каждый производственный участок должен иметь стандартизированное описание команд, которыми он может управляться, и данных, которые могут быть получены с него. При наличии подобного описания любой сторонний разработчик сможет обеспечить интеграцию системы управления данным производственным участком со своей системой. В настоящее время CBA, к сожалению, пока не получила широкого распространения.

В силу того, что технологии взаимодействия ПЛК с системами верхнего уровня уже достигли высокой степени стандартизации, обмен данными между ПЛК через вышестоящую систему легко организовать даже в том случае, если они выпущены разными фирмами. Обычно в качестве вышестоящей системы выступает SCADA. При этом могут использоваться любые технологии связи ПЛК и SCADA: OPC, HTTP, FTP и др. Но не

следует забывать о серьёзном недостатке такого подхода: в обмене данными участвует персональный компьютер - элемент, гораздо менее надёжный, чем ПЛК. Если он «зависнет» или выйдет из строя, обмен данными прервётся. Поэтому приходится либо тратить дополнительные средства на резервирование рабочей станции SCADA, либо мириться с достаточно высокой вероятностью прерывания обмена данными.

Рассмотрим теперь более подробно технологии, с помощью которых SCADA может получить данные с контроллера. Самый простой способ - использование специальных драйверов, разработанных поставщиками оборудования или SCADA. Хотя этот способ и позволяет достичь наибольших скоростей обмена, в настоящее время он применяется довольно редко из-за того, что драйверы для связи с оборудованием, как правило, создаются для узкого набора SCADA, остальные же системы оказываются «за бортом». Самой же популярной технологией связи ПЛК и любых компьютерных приложений, в том числе SCADA, является технология OPC. Первоначально эта аббревиатура расшифровывалась как OLE for Process Control, поскольку первоначально в основе обмена данными лежала технология OLE (Object Linking and Embedding) компании Microsoft. Соответственно, OPC в подавляющем большинстве случаев использовалась только на компьютерах под управлением операционной системы Windows.

В последние годы сформировалась тенденция к большему разнообразию технологий и платформ, участвующих в обмене данными с ПЛК, поэтому ассоциацией OPC Foundation было принято решение о разработке новой спецификации - OPC Unified Architecture (OPC UA), которая не ограничивала бы пользователей и разработчиков в использовании той или иной технологии для обмена данными. В OPC UA описывается только информация, которая должна передаваться между разными уровнями управления, при этом выбор способа передачи информации остаётся за разработчиком системы [9]. С отказом от преимущественного использования OLE связан и тот факт, что термин "OPC" теперь обычно используется без расшифровки.

Однако до тех пор, пока OPC UA не получил широкого распространения, наиболее популярными стандартами обмена данными между ПЛК и компьютерными приложениями остаются OPC Data Access (OPC DA) и OPC Historical Data Access (HDA). Первый используется для обмена данными с контроллером в реальном времени, второй - для получения архивных данных с контроллера или другого устройства, например, счётчика электроэнергии. Широкое распространение OPC DA позволило фирмам-поставщикам SCADA отказаться от разработки драйверов для связи с устройствами, так как большинство крупных производителей ПЛК и систем управления поддерживают стандарт OPC и выпускают OPC-серверы для своего оборудования. Это же обстоятельство даёт возможность забыть о том, что контроллеры могут подключаться к компьютеру с использованием самых разных технологий и сетевых протоколов. Для SCADA одинаково выглядят контроллер, доступ к которому производится по сети PROFIBUS (на основе RS-485), по PROFINET (на основе Ethernet) и устройство, подключенное по интерфейсу «токовая петля», поддерживающее закрытый протокол производителя.

Современные контроллеры всё чаще оснащаются интерфейсом Ethernet (или совместимой с ним разновидностью Industrial Ethernet, лучше удовлетворяющей требованиям эксплуатации на промышленном производстве). В состав встроенного программного обеспечения включаются HTTP-серверы, а также серверы и клиенты FTP. Если HTTP используется в основном для диагностики контроллера с помощью Web-браузера, то FTP может применяться для передачи блоков данных в контроллер или из него. Поэтому именно FTP иногда рассматривается как возможная альтернатива OPC. По мнению авторов, применение FTP выглядит оправданным только в том случае, если нужно обмениваться большими объёмами данных относительно нечасто. Использование FTP для постоянного обмена данными с высокой скоростью (для чего следует применять либо OPC

DA, либо драйвер) сопряжено с необходимостью передавать значительные объёмы служебной информации, что затрудняет быстрый обмен.

Нельзя забывать, что SCADA и другие компьютерные приложения, работающие в составе АСУТП, должны получать данные не только с ПЛК, но и с самых разных дополнительных устройств: принтераппликаторов, сканеров штрих-кодов, считывателей RFID, устройств контроля качества, различных лабораторных устройств и т.д. Обычно такие устройства оснащаются коммуникационными портами стандартов RS-232 или Ethernet. В отличие от производителей ПЛК, их производители ещё не пришли к использованию единой технологии обмена информацией наподобие OPC. Поэтому при интеграции устройств в систему управления каждый раз приходится решать новые задачи. Некоторые устройства, например сканеры штрих-кодов, передают все необходимые данные в виде последовательности ASCII-символов, поэтому получение таких данных не представляет особого труда. Другие устройства, например различные сканеры дефектов, функционируют совместно со специальным ПО. Информацию о работе устройства такое ПО обычно заносит в специальную базу данных. Впоследствии информация считывается из базы данных или экспортируется в текстовый или xml-файл обмена и считывается уже из него. В любом случае, для получения доступа к информации о работе устройства необходимо или создание скриптов различной сложности в SCADA, или разработка специальных программ-коннекторов, считывающих информацию из базы данных или файла обмена и передающих эту информацию в SCADA.

Для хранения данных в SCADA используются два класса СУБД - обычные реляционные СУБД (РСУБД) и СУБД реального времени (СУБД РВ). Задача СУБД реального времени состоит в обеспечении возможности интенсивной записи данных о ходе технологического процесса, хранения этих данных в течение некоторого промежутка времени и дальнейшей передачи в РСУБД. Необходимость их использования обусловлена тем, что распространённые РСУБД не позволяют записывать данные настолько часто, как это требуется в системах управления реального времени. В составе многих SCADA имеется собственная реализация СУБД РВ, которая, как правило, используется в качестве буфера. Существуют также СУБД реального времени, выпускаемые независимыми разработчиками (например Berkley DB, принадлежащая в настоящее время Oracle), но в SCADA они используются нечасто.

Большой интерес представляют СУБД РВ, входящие в специализированные системы сбора, хранения и анализа данных о ходе технологического процесса - Historian компании Wonderware (ранее известный как Industrial SQL Server), Plant Data Archive, входящий в состав Simatic IT Historian (Siemens) и аналогичные продукты других разработчиков. Подобные системы могут не только выступать в роли промежуточного звена, выполняющего сбор, первоначальную обработку и передачу данных в РСУБД, но и хранить данные в течение достаточно длительного времени. Фактически, как и в РСУБД, максимальный интервал хранения определяется только аппаратными характеристиками сервера баз данных. Для долговременного хранения данных о ходе технологического процесса также используются широко распространённые СУБД, такие как Microsoft SQL Server, Sybase SQL Anywhere, Postgres, Paradox, dBase [10].

## **ИНТЕГРАЦИЯ С MES-УРОВНЕМ**

Определённые особенности имеет и задача интеграции приложений управления производственными процессами [11]. После того, как данные попали в SCADA, доступ к ним для чтения и изменения значительно упрощается, но появляется новый барьер. Дело в том, что компьютерные информационные системы, работающие на разных уровнях управления, в своей работе используют различную терминологию, оперируют интервалами времени разной длины и, наконец, имеют различные цели. Если технологический процесс представляется

примерно одинаково и в программе ПЛК, и в SCADA, то MES-система уже использует упрощённую модель процесса, а ERP - ещё более упрощённую. Если период обработки данных в SCADA должен быть сопоставим со скоростью протекания технологического процесса (от нескольких секунд до нескольких минут), то MES-системы обрабатывают данные за время рабочего цикла или смены, а ERP-системы оперируют данными за месяц или полугодие.

Всё это приводит к необходимости «перевода» информации, пересекающей границу уровня управления, в формат, поддерживаемый системой, с которой производится обмен. Более предпочтительно использование стандартных форматов передачи информации, поддерживаемых всеми системами. Применение такого формата означает, что системы, участвующие в обмене, принимают некоторую модель производства, которая, может быть, и отличается от их собственных моделей. Предложено несколько таких моделей, но лучше всего разработана модель, описанная в серии стандартов ISA-95 [12, 13].

Говоря об ISA-95 следует упомянуть о его предшественнике - стандарте ISA-88. В ISA-88 определена иерархическая модель производственного оборудования, которую предлагается использовать при описании периодических (batch) технологических процессов. ISA-88 был благосклонно принят и производителями программного обеспечения систем управления, и пользователями-технологами. Поэтому, когда в конце 90-х годов появилась проблема обеспечения взаимодействия между корпоративными системами и системами производственного уровня (в первую очередь MES), организация ISA решила использовать опыт создания ISA-88 для разработки нового стандарта. Этим стандартом и стал ISA-95 [15].

В ISA-95 определено 4 уровня управления производственным предприятием. На первом уровне располагаются локальные системы управления (регуляторы, частотные преобразователи и т.д.) и интеллектуальные датчики, на втором - распределённые системы управления, системы управления периодическими процессами, а также SCADA. К третьему уровню относятся MES, лабораторные информационные системы (LIMS), системы складского хранения (WMS) и т.д. На четвёртом уровне расположены корпоративные информационные системы, в первую очередь ERP. Упоминается также уровень 0, к которому относится технологическое оборудование. Основное внимание в стандарте уделяется уровню 3 и взаимодействию между уровнями 3 и уровнем 4.

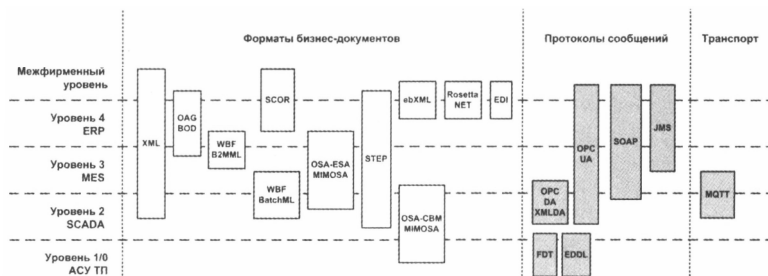
С точки зрения ISA-95 на уровне 3 решаются следующие задачи: планирование производства, управление производством, учёт материалов и энергозатрат, обеспечение качества, управление запасами, управление техническим обслуживанием. На уровне же 4 решаются такие задачи, как обработка заказов, расчёт стоимости продукции, управление поставками и управление отгрузкой. Решение этих задач требует обмена информацией четырёх классов: информация об определении (какие ресурсы необходимы для решения задачи), информация о возможностях (какие ресурсы доступны), информация о производственных планах (что и когда нужно сделать), информация о производительности (что и как было сделано). При этом под задачей может пониматься как производство продукции, так и техническое обслуживание, управление запасами и обеспечение качества. Следовательно, уровни 3 и 4 могут обмениваться информационными сообщениями 16 видов (4 класса информации, 4 возможные задачи для каждого класса).

Важной частью информационного обмена является обмен информацией о ресурсах. В ISA-95 также определяется иерархическая модель производственного предприятия. В отличие от модели ISA-88 эта модель не охватывает единицы оборудования, а описывает производственное предприятие начиная с самого верхнего уровня (т.н. уровень компании) и заканчивая производственными линиями в целом. Данная модель оборудования используется в дальнейшем при задании соответствующих информационных сообщений, например, такого: «Для производства доступна линия №2 цеха 35 Саратовского производственного объединения».

Несмотря на то, что изначально ISA-95 разрабатывался для решения задач вертикальной интеграции, его концепции используются и при обмене данными между MES и другими системами уровня 3. Так, в соответствии с рекомендациями MESA International, задачу планирования в настоящее время стремятся решить не средствами MES, а с помощью специальных систем - APS (Advanced Planning System). Для взаимодействия MES и APS вполне возможно использовать информационные сообщения, определённые в ISA-95.

Следует отметить, что сам по себе стандарт ISA-95 (как и ISA-88) не содержит спецификаций форматов обмена информацией, он лишь описывает концепции, которые можно использовать при создании таких форматов. Примером реализации концепций ISA-95 служит язык B2MML - набор XML-схем, предназначенный для взаимодействия любых систем с 3 и 4 уровня. ISA-95 - не единственный подход к интеграции систем на уровне MES, могут использоваться и другие стандарты: ISA-88 (BatchML), OAGiS, MIMOSA, STEP. Подробно эти стандарты (и некоторые другие) описаны в статье [13], а их место относительно иерархии ISA-95 показано на рис. 2.

**Рис. 2. Связь стандартов интеграции с уровнями ISA-95**



К сожалению, стандарты обмена информацией разработаны не в России и опубликованы на английском языке, что сдерживает их применение в нашей стране. В последнее время, однако, ситуация начинает меняться. В различных русскоязычных изданиях появляются материалы о стандартах интеграции, делаются доклады на отечественных конференциях. Российская рабочая группа MESA International ([www.mesarussia.ru](http://www.mesarussia.ru)) в данный момент ведёт работу по переводу на русский язык стандарта ISA-95.

Рассмотрим другие способы обмена информацией между MES и другими системами. Основной потребитель данных, хранящихся в SCADA - MES-системы. Практически все SCADA имеют встроенный OPC-сервер, поэтому OPC DA и OPC HDA остаются наиболее популярными технологиями обмена данными между SCADA и MES. К ним относятся: обмен данными через XML-, csv-файлы и другие структурированные текстовые файлы, передача данных через COM-интерфейсы SCADA, а также прямые запросы к базам данных, в которых SCADA хранит информацию.

Прямое обращение к базе данных технически реализуется довольно просто. В настоящее время все широко используемые СУБД поддерживают стандартные технологии доступа к данным: ODBC, JDBC, ADO. Основная проблема заключается в другом: прямое обращение к базе данных SCADA имеет смысл только в том случае, когда известна используемая ею схема данных. Если на уровнях MES и ERP ещё встречаются примеры использования стандартных схем данных (например PODS или APDM в нефтегазовой отрасли), то производители SCADA подобные схемы практически не используют, и, кроме того, редко сопровождают свои продукцию описанием схемы данных. Поэтому нередко случаи, когда разработчикам, выполняющим интеграцию, приходится делать такое описание самим. В результате имеется большая вероятность получить решение, накрепко привязанное к текущим версиям SCADA и MES, так как любые изменения схемы данных, выполненные



производителем, делают его неработоспособным. Нельзя забывать и о том, что в SCADA или MES могут быть реализованы механизмы поддержания целостности данных, поэтому при прямом обращении целостность может быть нарушена.

Всё это позволяет сделать вывод о том, что прямая работа с базами данных систем должна использоваться только тогда, когда применение других способов обмена невозможно. При выборе подобного подхода не стоит забывать и о том, что прямая связь на уровне БД не гарантирует качества и целостности данных.

## **ИНТЕГРАЦИЯ БИЗНЕС-ПРИЛОЖЕНИЙ**

Отдельного внимания требует вопрос интеграции корпоративных бизнес-приложений. Приложения этого уровня называют часто промежуточными (middleware). В этот класс входят системы бизнес-аналитики, отчётности, поддержки принятия решений и т.п. На уровне данных и даже операций они активно взаимодействуют с другими системами предприятия всех уровней - MES, ERP, и т.д. Важность данного интеграционного уровня уже неоднократно отмечалась [14], поэтому не будем повторяться о важности темы, а сосредоточимся на существующих подходах и стандартах.

Вопросов, связанных с интеграцией на этом уровне достаточно много, существует даже международный Консорциум по интеграции (<http://www.integrationconsortium.org/>). Этот консорциум позиционирует себя как площадка для общения всех специалистов, заинтересованных в решении или поиска путей решения задач, направленных на интеграцию информационных систем.

В данной статье мы не берёмся обсуждать интеграцию как растянутый во времени процесс, а сфокусируемся лишь на имеющихся в нашем распоряжении технических средствах. Википедия определяет понятие «Enterprise application integration» как совокупность программного обеспечения и принципов построения автоматизированных систем, направленных на интеграцию компьютерных приложений. Не смотря на то, что на данную тему уже написано огромное множество работ и информационных материалов, попробуем, всё-таки, ввести некоторую классификацию, чтобы было легче ориентироваться при выборе оптимального решения. Это чрезвычайно важно, т.к. с увеличением сложности систем и набора обрабатываемых данных, интеграция становится всё более и более затратным проектом, и одна из тенденций направлена на то, чтобы сделать этот процесс по возможности простым и эффективным. Появился даже термин «бережливый промежуточный слой» (lean middleware).

Все решения на данном уровне, с нашей точки зрения, можно разделить на несколько групп:

- интеграция на уровне обмена документами
- интеграция через унификацию моделей данных и синхронизацию непосредственно в СУБД отдельных приложений
- интеграция на уровне нормативно-справочной информации (НСИ)
- интеграция через обмен сообщениями
- работа с универсальным промежуточным контейнером сервисов, т.н. сервисная шина (ESB)

Отметим, что, как правило, системные интеграторы пропагандируют последний подход как самый правильный, но на практике, в особенности для небольших компаний, он применяется редко из-за высокой стоимости владения таким решением.

### **Обмен документами**

Интеграция на уровне обмена документами достаточно очевидна. Одно приложение формирует некий документ, как правило, это либо текстовый файл, либо csv (данные,

разделённые заданным символом), либо xml, реже xls (Microsoft Excel). Или любой другой согласованный формат. Приложение-получатель в ручном или автоматическом режиме импортирует его и обрабатывает хранящиеся данные. Как правило, для обмена используются общие сетевые папки, иногда электронная почта, сервисы FTP. Каких-либо рекомендаций по организации интеграции на таком уровне дать невозможно, стандартов документов (в т.ч. на основе xml) настолько много, что выбор одного из них является большой редкостью, и чаще всего просто определяется свой новый формат на этапе разработки проектных решений по интеграции систем.

Несомненным достоинством такого подхода является его простота реализации и универсальность - появляется возможность использовать в едином комплексе программ устаревшие приложения, ввод данных может быть реализован даже через эмуляцию операторского ввода. Именно по этой причине такой вариант интеграции является самым распространённым, часто подобный обмен запускался как временная схема и работал в таком режиме годами. Недостатком же метода является сложность управления потоком разрозненных документов между приложениями, синхронный переход на новые версии при расширении состава передаваемой информации, сложность администрирования прав доступа к документам и т.д.

Рекомендуем использовать такой подход для обмена данными между отдельными стоящими серверами, направленными на решение отдельной бизнес-задачи (например, комплексы моделирования), сбора консолидированной отчётности и т.п., где интеграция не относится к разряду критичных для бизнеса (business critical).

### **Интеграция на уровне модели данных**

Интеграция на уровне данных, хранящихся в СУБД различных систем, строится либо на базе специализированных программных систем, либо с использованием стандартизованных схем данных и использования единой СУБД или организации реплики штатными средствами. Стандартные инструменты оперируют с данными независимо от типа источника или его структуры. Существует множество стандартных методов доступа к данным, хранящимся в реляционных СУБД, как правило предоставляемых производителями вместе с программным обеспечением. Эти стандарты широко известны: ADO.NET, JDBC, ODBC, OLE DB, XQuery, Service Data Objects (SDO) [16].

Данные извлекаются из реляционных баз данных, мейнфреймов, приложений, XML, сообщений и даже из документов, таких как документы Microsoft Word, PDF или электронные таблицы Microsoft Excel. Далее, для получения «единого взгляда» на данные из всех систем, данные интегрируются и трансформируются для приведения к одним структурам вне зависимости от форматов систем-источников. Данные преобразуются в правильный формат в нужный момент времени и загружаются во все виды приложений и для пользователей, которым эти данные необходимы. Подобные решения есть практически у всех крупных производителей корпоративных систем, наиболее распространёнными являются решения от Oracle, SAP и Informatica.

Отметим, что доступ к данным из ERP-систем на уровне физических данных СУБД в большинстве случаев закрыт производителем. Для доступа должны использоваться специализированные инструменты и методы, например протокол C/Front от Microsoft и т.п. Поэтому, при разработке архитектурных решений по интеграции, необходимо обратить особое внимание на то, какие протоколы доступа разрешены разработчиком в системе.

Интеграционные решения на уровне обмена данными могут быть построены и по принципу оверлейных сетей на основе протоколов обмена peer-to-peer, таких как, например, JXTA (Juxtapose) [18]. На практике такие решения применяются достаточно редко, хотя в некоторых случаях, особенно связанных с заказной разработкой системы управления предприятием, могут быть достаточно эффективны. Обмен данными по принципу «точка-

точка» требует существенно меньших аппаратных мощностей по сравнению с централизованными интеграционными моделями, и, как следствие, обеспечивает более низкую стоимость владения системой, в некоторых случаях повышает надёжность, т.к. в системе отсутствуют узкие места.

Существенную поддержку при реализации интеграции на уровне СУБД может оказать работа со стандартной моделью данных. Такие модели можно найти во многих отраслях. Это CIM (Common Information Model) в электроэнергетике [19], PODS (Pipeline Open Data Standard) или его российская реализация ОСМД (открытая стандартная модель данных) в нефтегазовой отрасли [20], APDM (ArcGIS Pipeline Data Model) и др.

На уровне горизонтальной интеграции, как один из наиболее важных, необходимо отметить серию стандартов, относящихся к задаче управления жизненным циклом изделия [21] (PLM, Product Lifecycle Management) - ГОСТы Р ИСО серий 10303 и 13584. В них описываются стандарты электронного представления данных об изделии, они направлены на обеспечение механизмов, позволяющих передавать данные об изделиях и библиотеках деталей независимо от того, в какой прикладной системе созданы или используются эти данные. Данные стандарты описывают как непосредственно модель данных, так и стандарт языка EXPRESS, предназначенного для работы со структурой данных об изделиях. Интерфейс работы с данными описывается в «ГОСТ Р ИСО 10303-2-2001. Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 22. Методы реализации. Стандартный интерфейс доступа к данным».

### **Интеграция на основе единой НСИ**

Наиболее очевидный способ обеспечения связи между данными в различных приложениях - построить все системы на базе единой нормативно-справочной информации, НСИ (MD, Master Data). Напомним, что нормативно-справочной считается условно-постоянная информация предприятия. Правильно спроектированная система НСИ обеспечит лёгкую интеграцию информации из любых систем, т.к. исключит проблемы, связанные с низкой целостностью данных. Однако на практике так бывает редко, и каждая система имеет свой состав справочников НСИ. Задача обеспечения их взаимосвязки достаточно сложна, и не только с точки зрения программной реализации. Часто проблема заключается в том, что различные системы оперируют с различной структурной моделью производственной системы. Так, например, диспетчеру важны потоки и модель диспетчерской системы рассматривает систему параллельных трубопроводов как один, в то время как для ремонтной бригады принципиально их разделение.

Методики построения связанной корпоративной НСИ описаны в литературе достаточно подробно [22]. Различают полностью централизованную, частично централизованную, децентрализованную с консолидацией в единой системе, и т.п. схемы построения. В некоторых случаях даже целесообразно строить НСИ на основе древовидных структур каталогов типа DAP (Directory Access Protocol), например на основе LDAP (Lightweight DAP) или «полновесного» варианта X.500. Примером может служить хранение данных о пользователях в Microsoft Active Directory. Такой подход в реальных проектах применяется крайне редко, хотя, по мнению авторов, достаточно удобен для целого ряда специфических задач.

Класс специализированных систем, равно как и сама задача управления корпоративной НСИ, называется MDM (Master Data Management) [23]. В этом сегменте можно отметить решения от Oracle Data Integration Suite, SAP MDM, IBM InfoSphere MDM Server. Они все обеспечивают базовую функциональность систем управления НСИ: консолидация и хранение данных, управление структурой и составом справочников и классификаторов, изменениями данных, поддержание распределённой работы, формирование персонализированного представления НСИ пользователям, поиск и просмотр данных,

распространение изменённых данных, защита данных от несанкционированного доступа и т.д.

Но, возвращаясь к реальной практике, заметим, что интеграция на уровне НСИ в той или иной степени обязательно присутствует в реальных системах, но реализована она, как правило, на перекодировочных таблицах. Часто в структуру записи НСИ ещё разработчиком включено поле для хранения кода внешней системы, которое может использоваться для задач интеграции. Такой подход, даже при децентрализованном управлении справочниками, зачастую полностью оправдан, особенно если реализованы эффективные средства контроля целостности данных.

Говоря об интеграции на уровне НСИ отметим ещё и такой подход, как максимальное использование существующих стандартных справочников и классификаторов, таких как, например, ОКУД (Общероссийский классификатор управленческой документации) или справочник регионов ГИБДД. Изначально заложенная ориентация на существующие государственные или отраслевые справочники позволит в последствии существенно сократить усилия, направленные на формирование единой корпоративной НСИ.

### **Сервис-ориентированный подход**

Но наиболее перспективной на сегодняшний день считается интеграция разнородных приложений на уровне операций, а не только данных, т.е. когда одно приложение выполняет для другого некую функцию, предоставляет информационный сервис. Такой подход получил название сервис-ориентированной архитектуры (SOA - Service Oriented Architecture). По определению одного из членов рабочей группы W3C Web Services Architecture Working Group, «Сервис есть единица работы, выполненная поставщиком сервиса для достижения конечного результата, необходимого потребителю сервиса». Фактически технология SOA пришла на смену таким технологиям, как DDE (Dynamic Data Exchange) или RPC (Remote Procedure Call), которые тоже были направлены на выполнение некоторых функций одних приложений в рамках задач, выполняемых другими приложениями.

В сервис-ориентированной архитектуре приложения интегрируются на базе единого контейнера (или, как его ещё называют, брокера) сервисов. По типу интеграции такое взаимодействие относится к интеграции на базе шины (bus) (выделяется ещё интеграция через концентраторы, hub-and-spoke), которая обеспечивает транзакции, преобразование данных, сохранность обращений. Решения, реализующие такой принцип, называют сервисной или, иногда, интеграционной шиной предприятия (ESB - Enterprise Service Bus) [17]. Отметим, что ESB сама по себе не включает функциональность SOA, но является инструментом, обеспечивающим его. Из существующих решений в нашей стране наиболее известны SAP XI/PI (Exchange Infrastructure/Process Integration), BizTalk от Microsoft, WebSphere от IBM, JBoss от RedHat, Celtix от консорциума ObjectWeb.

Классическая корпоративная сервисная шина поддерживает Web-сервисы, реализуя протокол SOAP (Simple Object Access Protocol, Простой протокол доступа к объектам) и используя язык WSDL (Web Services Description Language, Язык описания Web-сервисов) и спецификацию UDDI (Universal Description, Discovery and Integration, Универсальное описание, обнаружение и интеграция). Многие корпоративные сервисные шины также поддерживают другие модели обмена информацией, включая гарантированную доставку и «публикацию и подписку» (publish and subscribe) [26].

Дальнейшее развитие сервис-ориентированная архитектура получила в виде, так называемых, Web-сервисов. Сразу предостережём от путаницы понятий, т.к. так же называют и Веб-ресурсы, предоставляющие определённые сервисы, именно с ними связаны такое понятие, как «Intellegent Web Service» со своими языками реализации DAML и пр. Эти спецификации не относятся к теме настоящей статьи, спецификации UDDI и WSDL иногда для избежания путаницы называют промышленными стандартами. Кроме этого, встречается

термин UAN (Universal Application Network), а Web-сервисы рассматриваются как «транспортный уровень» в архитектуре UAN.

Среди спецификаций в области управления Web-сервисами можно выделить язык BPEL (Business Process Execution Language), сокращение от WS-BPEL (Web Services Business Process Execution Language). Другие стандарты - как, например, BPML (Business Process Modeling Language, Язык моделирования бизнес-процессов), WSCI (Web Service Choreography Interface, Интерфейс взаимодействия Web-сервисов), XPDЛ (XML Process Definition Language, Язык описания процессов) и BTP (Business Transaction Protocol, Протокол бизнес-транзакций) - обладают определёнными техническими достоинствами, однако, не поддерживаются большинством поставщиков.

Язык BPEL [27] объединяет возможности языка WSFL (Web services flow language, Язык организации потоков Web-сервисов), разработанного компанией IBM, и языка XLANG, используемого в Microsoft BizTalk Server. BPEL включает WSFL для поддержки графоориентированных процессов, а XLANG - для поддержки структурных конструкций для процессов. Таким образом, BPEL предназначен для поддержки реализации бизнес-процессов любой сложности, а также для описания интерфейсов бизнес-процессов.

Язык BPEL неразрывно связан со спецификациями WS-Coordination (Координация Web-сервисов) и WS-Transaction (Транзакции Web-сервисов), которые были определены для совместного использования с BPEL и разработаны для координации транзакций и процессов. Так, в спецификации WS-Coordination описываются стандартные механизмы создания и регистрации протоколов транзакций, которые координируют выполнение распределённых операций в среде Web-сервисов. С помощью спецификации WS-Transaction можно отслеживать успех или неудачу каждого отдельного скоординированного действия в бизнес-процессе, задавать гибкую модель транзакций, которая обеспечивает целостность и надёжность операций в распределённой среде Web-сервисов и позволяет бизнес-процессам обрабатывать сбои в ходе выполнения.

И было бы ошибкой не отметить такой очевидный факт, что эффективное построение интегрированной среды на основе SOA-архитектуры невозможно без наличия грамотно организованной структуры НСИ [24].

### **Обмен сообщениями**

Интеграция через обмен сообщениями [25] предполагает, что различные приложения могут передавать данные и команды по сети по принципу «отправил и забыл» и вернуться к выполнению текущей задачи. Связующее программное обеспечение, ориентированное на обмен сообщениями, называется MOM (Message Oriented Middleware). В такой технологии стандартными механизмами являются решения на основе JMS (Java Message Service), службы Microsoft Message Queuing

(MSMQ), стандарты Web-служб, поддерживающие асинхронный обмен сообщениями, например, WS-ReliableMessaging, такие инструменты как JAXM (Java API for XML Messaging) от Sun Microsystems и WSE (Web Server Extensions) от Microsoft. Выделяют синхронный и асинхронный обмен сообщениями, последний используется гораздо шире в силу более широких возможностей.

Модель интеграции на основе обмена сообщениями поддержана всеми основными идеологами. Спецификация WS-Reliability, наделяющая Web-службы способностью обмениваться сообщениями в асинхронном режиме с гарантированной доставкой без дубликатов, утверждена консорциумом OASIS. Из других протоколов отметим ebXML (Electronic Business using XML), которая появилась ещё до SOAP и WSDL, ebMS (Electronic Business Message Service), разработанная консорциумом OASIS в 2002 году, и оперирующая не только с XML, но и с бинарными сообщениями. Есть множество спецификаций для Java, например, JBI (Java Business Integration). С целью расширения концепции Веб-служб сформирована

серия спецификаций, имеющих названия WS-\*. Это WS-Reliability, WS-ReliableMessaging, и др. К сожалению, регулярно появляются новые конкурирующие версии, что приводит к неразберихе в стандартах.

## **ИНТЕГРАЦИЯ С ВНЕШНИМИ СИСТЕМАМИ**

Принимая во внимание всё большее развитие кооперативного характера бизнеса, что выражается в широком распространении таких методов как технологии EDI (Electronic Data Interchange) [29], объединённого производства c-MES (Collaborative MES) [30] и др., при построении интеграционной среды важно не забывать и об обязательной поддержке форматов взаимодействия с внешними по отношению к рассматриваемой информационными системами.

В технологии EDI наибольшее распространение получили стандарты ANSI X-12 и UN/EDIFACT. На базе последнего, в частности, построена широко известная банковская система обмена информацией SWIFT. В целом, технология EDI оказалась достаточно удачной и получила достаточно широкое применение и поддержку практически всех крупных производителей программного обеспечения. Спецификация, ориентированная на работу в среде интернет, через защищённые протоколы, получила название Web EDI.

В 2000 году несколько российских компаний объединились и выпустили стандарт обмена коммерческой информацией на основе XML - CommerceML [31], который, по заверениям его авторов, позволяет существенно снизить затраты на организацию информационного взаимодействия за счет унификации обмена коммерческой информацией между различными организациями. Информация по стандарту доступна на официальном сайте проекта [www.commerceml.ru](http://www.commerceml.ru).

Говоря об EDI необходимо упомянуть и стандарт IDoc (Interchange Document), который раньше был популярен у разработчиков ERP-систем, в частности, это встроенный механизм обмена коммерческими документами системы SAP R/3. Принципиальное отличие этого формата в том, что он построен не по теговой схеме, как XML, а использует таблицы с данными и мета-данными. Кроме этого, IDoc работает с таким понятием, как сессия, отражающим, что в данный момент происходит с документом. В настоящее время в связи с развитием технологий EDI стандарт IDoc как обменный формат практически не используется, хотя внутренне применение в системе SAP R/3 сохранилось.

Но для локальных задач, особенно в рамках одной организационной структуры, чаще всего, всё-таки, используется обмен текстовыми (или XML) файлами. Достоинства и недостатки рассматриваемых схем примерно такие же, как и рассмотренные в случае интеграции на базе обмена документами между корпоративными приложениями. В качестве транспортного механизма чаще всего используется электронная почта, реже FTP-серверы, классические веб-сервисы на практике в нашей стране применяются крайне редко.

Для интеграции с офисными приложениями, такими как Microsoft Office, Star Office, Open Office и т.д. используется, как правило, либо прямое формирование отчётных документов в нужных форматах, либо специальные протоколы на базе XML, такие как, например, IBF (Information Bridge Framework) [28], который представляет собой набор компонентов и инструментов для создания продуктов, посредством веб-сервисов соединяющих корпоративные информационные системы с офисными приложениями.

## **ВЫВОДЫ**

Попытка описать возможные стандарты и технологии интеграции производственных информационных систем может показаться кому-то попыткой объять необъятное. Тем не менее, авторы статьи постарались, основываясь на личном опыте и опыте своих коллег, выбрать и описать лучшие и наиболее популярные из них. Информация о способах

интеграции особенно актуальна именно в нашей стране, где почти на каждом крупном заводе имеется собственный «зоопарк» систем управления разных производителей и годов выпуска. Время лоскутной автоматизации прошло, и эффективно управлять производством в современных условиях можно только в том случае, когда информация с одних уровней управления принимается во внимание и используется на других уровнях. Только так можно организовать гибкое современное производство, способное выпускать продукцию в нужном количестве, с нужной стоимостью, в заданное время.

Говоря об интеграции приложений, в особенности уровня управления предприятием, кроме прочего, необходимо иметь в виду, что задача интеграции тесно связана с таким понятием, как архитектура предприятия. Последняя, кроме понятий миссии, стратегии и бизнес-модели оперирует и с таким понятием, как системная архитектура, т.е. с общими правилами построения комплекса приложений, данных и оборудования, которые, впоследствии, не поддаются или с трудом поддаются изменению. Поэтому любые ошибки, допущенные на стадии концептуального проектирования системной архитектуры, окажут существенное влияние и на задачи интеграции систем и приложений.

Для полноты изложения отметим две основные методологии построения архитектуры предприятия: GERAM - Generalized Enterprise Reference Architecture and Methodology (построенная на основе схемы CIMOSA - Open System Architecture for CIM) и TOGAF - The Open Group Architecture Framework [32]. Оба подхода близки по своей сути, хотя и оперируют с различными модельными подходами. В настоящее время с учётом методологии GERAM разрабатывается ГОСТ «Системы промышленной автоматизации. Требования к архитектуре и методологии эталонных предприятий. Разработка ГОСТ Р. Прямое применение МС с дополнением - EQV (ISO 15704:2000, ISO 15704:2000/Amd.1:2005)» на основе международного стандарта ISO 15704:2000 «Системы промышленной автоматизации. Требования к архитектуре эталонных предприятий и методологии» («Industrial automation systems - Requirements for enterprisereference architectures and methodologies»), в который GERAM входит как приложение. Перевод самого стандарта на русский язык доступен в национальном фонде технических нормативных правовых актов Белоруссии.

Знание и понимание принципов построения архитектуры предприятия необходимо, т.к. задача интеграции не решается одними лишь техническими средствами. Реальное предприятие - живой организм, в котором постоянно происходят изменения, добавляются новые модули, появляются и исчезают бизнес-процессы. А интеграция, основанная на связывании нестандартизованного набора данных, будет работать лишь до следующего изменения в системе. Интеграционные решения и подходы должны быть неотъемлемой частью общей архитектуры предприятия.

## **Список литературы**

1. Integration Technologies for Industrial Automated Systems. CRC, 2006. 600 p.
2. Merz H., Building Automation: Communication systems with EIB/KNX, LON und BACnet. Springer, 2009
3. Ицкович Э.Л. Методы рациональной автоматизации производства. М.: Инфра-Инженерия, 2009. 256 с.
4. Денисенко В.В., Компьютерное управление технологическим процессом, экспериментом, оборудованием, М.: Горячая Линия - Телеком, 2009 г.
5. Федоров Ю.Н., Справочник инженера по АСУТП. Проектирование и разработка, М.:Инфра-Инженерия, 2008
6. The Industrial Communication Technology Handbook. CRC Press, 2005. 936 p.
7. Boyer S., SCADA: Supervisory Control and Data Acquisition, NY, 2004
8. PROFINET System Description (April 2009). PROFIBUS Nutzerorganization e.V., 2009. 52 p.

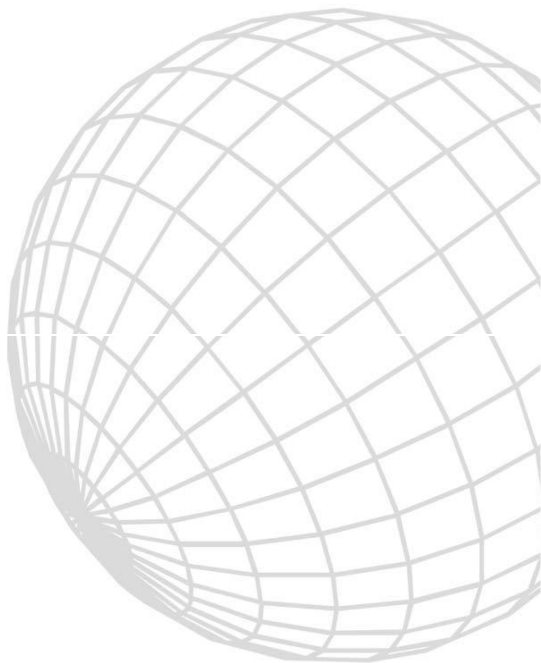
9. Mahnke W., Leither S., Damm M. OPC Unified Architecture. Springer, 2009. 339 p.
10. The Industrial Information Technology Handbook. CRC Press, 2005. 1936 p.
11. Halevi G. Handbook of Production Management Methods. Butterworth Heinemann, 2001.
12. ISA-95: The Enterprise-Plant Link to Achieve Adaptive Manufacturing, White Paper 16, MESA International 2005
13. Решетников И. и др. Стандарты интеграции многоуровневых информационных систем // Автоматизация в промышленности. 2009. № 9. С. 23-27.
14. Tamworth R., Westlund J., Интеграция MES и ERP – будущее производства? // Рациональное управления предприятием. 2006. № 1. С. 62-63
15. <http://www.isa-95.com/subpages/technology/isa-95.php>
16. Дунаев С., Доступ к базам данных и техника работы в сети, М.:Диалог-МИФИ, 2007
17. Шаппел Д., ESB – Сервисная Шина Предприятия, СПб:БХВ-Петербург, 2008
18. Brookshier D. и др. JXTA: Java P2P Programming, Sams Publishing
19. IEC 61970-301. Energy management system application program interface (EMS-API) – Part 301: Common information model (CIM) base.
20. Серебряков А.М. и др., Электронная исполнительная документация «как-построено» – шаг к новому качеству магистральных газопроводов // Газовая промышленность, 2008, № 7
21. Ковшов А.Н. и др., Информационная поддержка жизненного цикла изделий машиностроения. Принципы, системы и технологии CALS/ИПИ, М.:Академия, 2007
22. Колесов А., Технология управления НСИ корпоративного уровня // PC Week/RE, 2005, №18
23. Berson A., Dubov L., Master Data Management and Customer Data Integration for a Global Enterprise, McGraw-Hill, 2007
24. Гулько Д., Единая система НСИ - основа сервисно-ориентированной архитектуры // Intelligent enterprise, 2006, №22
25. Хоп Г., Вульф Б. Шаблоны интеграции корпоративных приложений. М.: Вильямс, 2007.
26. Juric M. и др. SOA Approach to Integration. XML, Web services, ESB, and BPEL in real-world SOA projects. PACTK, 2007.
27. Juric M., Business Process Execution Language for Web Services: BPEL and BPEL4WS, PACTK, 2004
28. Бурьески Ж. Интеграция офисных и корпоративных приложений // Intelligent Enterprise. 2005. № 21 (130). С. 17-21.
29. Календарев А., Введение в EDI, <http://www.citforum.ru/internet/articles/xmlledi.shtml>
30. Модель MESA для системы управления объединённым производством (с-MES) / в сб. MES - теория и практика, вып. 1 (2009), М.:Российская рабочая группа MESA, С. 5-26 / [www.mesarussia.ru](http://www.mesarussia.ru)
31. Ртищева Е., CommerceML – стандарт обмена коммерческой информацией в формате XML // Системный администратор, № 10, 2002
32. Goikoetxea A., Enterprise architectures and digital administration: planning, design and assessment, World Scientific Pub., 2007

Авторы признательны Российской рабочей группе MESA International за предоставленные материалы.  
 Решетников Игорь Станиславович – кандидат технических наук, заместитель начальника управления автоматизации, информатизации, телекоммуникаций и связи ООО «Газпром центрремонт»  
 Козлецов Алексей Павлович – кандидат технических наук, ведущий программист ООО «АМастер»





# Глоссарий терминологии MESA



## ГЛОССАРИЙ

MES	Manufacturing Execution System	Автоматизированная система управления процессами производства	Класс автоматизированных систем, предназначенных для контроля и оптимизации производственной деятельности предприятий. В иерархии управления MES занимают промежуточный уровень между ERP-системами и АСУ ТП	MESA WP
ISA	International Society of Automation	Международное общество по автоматизации	Организация по стандартизации управления производством и производственной автоматизации	www.isa.org
ISA-88	Standard Specification for batch processing industries	«Управление периодическим производством»	Стандарт ISA, определяющий структуру данных и информационные модели для управления периодическим производством.	ANSI/ISA88
ISA-95	Enterprise – Control System Integration specification	"Интеграция систем управления предприятием и технологическим процессом"	Стандарт ISA, определяющий структуру данных и информационные потоки для взаимодействия уровней управления производством и предприятием. Описывает рабочие процессы производственной деятельности предприятия.	ANSI/ISA95
SOA	Service Oriented Architectures	Сервисно-ориентированная архитектура	Подход к разработке программного обеспечения, в основе которого лежат сервисы со стандартизированными интерфейсами	MESA WP
MOM	Manufacturing Operations Management	Управление технологическим процессом; Оперативное управление производством	Методология, предназначенная для увеличения эффективности производства.	ГОСТ Р 50.1.031-2001 ГОСТ 3.1109-82 ANSI/ISA95
XML	Extensible Markup Language	расширяемый язык разметки	Рекомендованный World Wide Web Consortium стандарт	W3C Standard
B2B	Business to Business	взаимодействие между предприятиями		MESA WP
		сделка «бизнес-бизнес»	Электронная коммерческая сделка, сторонами которой являются юридические лица (предприятия, фирмы и т.д.)	ГОСТ Р 50.1.031-2001
B2C	Business to Customer	Взаимодействие между предприятием и покупателем		MESA WP
		сделка «бизнес-потребитель»	Электронная коммерческая сделка, сторонами которой являются юридическое лицо (предприятие, фирма и т.д.) - поставщик и физическое лицо - потребитель (покупатель)	ГОСТ Р 50.1.031-2001
B2M	Business to Manufacturing	Взаимодействие между бизнесом и производством	Определяет стандарты взаимодействия между MES и ERP системами	ANSI/ISA95
ERP	Enterprise Resource Planning	планирование и управление ресурсами предприятия	программное обеспечение, используемое для учета и планирования ресурсов организации	ANSI/ISA95

B2MML	B2M Markup Language	язык разметки связи бизнес-производство		ANSI/ISA95
KPI(s)	Key Performance Indicator(s)	ключевой показатель эффективности ТЭП (технико-экономический показатель)	показатель, поддающийся количественному измерению и считающийся наиболее важным для оценки эффективности деятельности предприятия	ANSI/ISA95
ANSI	American National Standards Institute	Американский национальный институт стандартов		www.ansi.org
OPC	Object Linking & Embedding (OLE) for Process Control	OLE для управления производственным и процессами	программная технология, предоставляющая набор объектов, используемых в автоматизации технологических процессов	www.opcfoundation.org
CAPA CAPAs	Corrective and Preventative Actions	корректирующее и предупреждающее действие	Действие, предпринятое для устранения причины обнаруженного и/или потенциального несоответствия или другой нежелательной ситуации.	ГОСТ Р ИСО 9000-2001
SCOR	Supply Chain Operations Reference model	Референтная модель операций в цепях поставок	Межотраслевой стандарт	Supply Chain Council WP
BPEL	Business Process Execution Language	язык на основе XML для формального описания бизнес-процессов и протоколов их взаимодействия	Межотраслевой стандарт	OASIS WP
BOD	Business Object Definition	Описание структуры бизнес-объектов	Совокупность описаний бизнес-объектов предприятия, их атрибутов и связей	ГОСТ Р 50.1.031-2001
ESB	Enterprise Service Bus	сервисная шина предприятия	обеспечивает взаимосвязь между различными приложениями по различным протоколам взаимодействия	MESA WP
MRPII (MRP II)	Manufacturing Resource Planning	Планирование производственных ресурсов	методологию, направленную на планирование в материальном и в денежном выражении	MESA WP
WBF	World Batch Forum	Ассоциация по технологии производства		www.wbf.org
WIP	Work-in-Progress/Work-in-Process	Незавершенное производство	Частично готовая продукция, не прошедшая предусмотренный технологией полный цикл производства.	MESA WP Налоговый кодекс РФ
MRP	Manufacturing Resource Planning	планирование производства	Совокупность программных средств и данных, обеспечивающая выполнение следующих функций: - объемное планирование и формирование графиков производства; - планирование по группам продукции для основных подразделений; - расчет производственных мощностей основных подразделений и определение мер, обеспечивающих соответствие мощностей объемам выпуска;	ГОСТ Р 50.1.031-2001

			- расчет и планирование потребностей в материалах и комплектующих с учетом графиков производства; - расчет сменно-суточных плановых заданий для подразделений и технологического оборудования (оперативное производственное планирование) и т.д.	
ROI	Return on Investment	рентабельность инвестиций		MESA WP
SCM	Supply Chain Management	управление цепочками поставок	Автоматизированная система, предназначенная для управления производственной цепочкой поставок. Включает управление внутренними ресурсами предприятия и весь цикл внешней логистики	MESA WP
SPC	Statistical Process Control	Системы статистического управления процессами	Предназначены для увеличения эффективности, возможностей и уменьшения стоимости производства	ГОСТ Р ИСО 11462-1-2007
SOAP	Simple Object Access Protocol	протокол обмена структурированными сообщениями	Межотраслевой стандарт	W3C Standard
ISO	International Standards Organization	Международная организация по стандартизации	Российскую Федерацию представляет Федеральное агентство по техническому регулированию и метрологии	ГОСТАНД АРТ РОССИИ
ID	(Device, Data ) Identification	Идентификатор (устройства, данных)		ГОСТ 34.340-91
LIMS	laboratory Information Management System	Лабораторные информационные менеджмент-системы		Проект ГОСТ Р Лабораторные информационные менеджмент-системы
IEC	International Engineering Consortium			
PLM	Product Lifecycle Management	Управление жизненным циклом продукции	Управление последовательными и взаимосвязанными стадиями производственной системы - от добычи сырьевых материалов или использования природных ресурсов до размещения (утилизации) отходов	ГОСТ Р ИСО 14040-99
MAF	Manufacturing Application Framework	Инфраструктура приложений предприятия	интегрированная среда, содержащая библиотеки классов и определяющая структуру данных информационно-управляющей среды предприятия	MESA WP
BI	Business Intelligence	бизнес-аналитика, корпоративный интеллект (по аналогии с искусственным интеллектом)	совокупность внедренных баз знаний, экспертных систем, автоматизированных систем управления, систем поддержки принятия решений, телекоммуникационных сетей и интернет-технологий предприятия	MESA WP

TCO	Total Cost of Ownership	общая стоимость владения	методика определения наилучшего соотношения цена/качество для основных средств. Критериями оценки являются стоимость приобретения, установки, администрирования, технической поддержки и сопровождения, модернизации, вынужденных простоев и других затрат	MESA WP
EAI	Enterprise Applications Integration	интеграция приложений предприятия	объединение в одну систему всех прикладных подсистем предприятия	MESA WP
MSA	Manufacturing Services Architecture			
MOCs	Manufacturing Operation Categories	Категории производственных операций	Стандарт определяет 4 основных категории производственных операций: Производство (изготовление) Тест качества Техническое обслуживание Инвентаризацию	ANSI/ISA95
SMART	Specific, Measurable, Acceptable, Realistic and Time-based	Точная спецификация, Измеримость, реалистичность, контролируемость по времени	Цель (задача) должна быть точно описана, измерима, реалистична и контролируема во времени	MESA WP
MDM	Master Data Management	Управление Нормативно-Справочной информацией	Нормативно-справочная информация содержит стандарты, требования, правила, положения и прочую данные, нормирующие и систематизирующие деятельность компании	ГОСТ 34.003-90 MESA WP
MSB	Manufacturing Service Bus	Производственная сервисная шина	Часть архитектура SOA. Обеспечивает взаимодействие между различным ПО в производственной сети предприятия.	MESA WP OASIS Reference Model for Service Oriented Architecture
DHR	Device History Records	Записи истории устройства	Набор записей, описывающих устройство и историю его состояний. Включает: серийный номер, состав устройства (запчасти), историю обследований, историю отказов, историю ТО и др. данные	MESA WP
MIMOSA	Machine Information Management Open Systems Alliance		Некоммерческая организация. Цель деятельности: разработка и продвижение открытых стандартов для производства и ТОиР в обрабатывающей промышленности	<a href="http://www.mimosa.org">www.mimosa.org</a>
MIS	Manufacturing Information Systems	Производственная информационная система		ANSI/ISA95 ANSI/ISA88
RFP	Request For Proposal	заявка на подряд (запрос предложений)	Документ, используемый заказчиком в качестве средства для объявления о своих намерениях выступить в качестве потенциального покупателя	ГОСТ Р ИСО/МЭК 12207-99

API(s)	Application Program Interface(s)	Прикладной программный интерфейс (интерфейс прикладного программирования)	Интерфейс между прикладным программным средством и прикладной платформой, через который обеспечиваются все услуги	ГОСТ Р ИСО/МЭК ТО 10000-3-99
PC(s)	Personal Computer(s)	Персональная ЭВМ	Настольная микро-ЭВМ, имеющая эксплуатационные характеристики бытового прибора и универсальные функциональные возможности	ГОСТ 15971-90
SCADA	Supervisory Control and Data Acquisition	система диспетчерского управления и сбора данных		ГОСТ 4.186-85
SON	Statement of Need	Описание задач	Документ, содержащий постановку задачи, цели и ожидаемые результаты от внедрения системы	MESA WP
EDI	Electronic Data Interchange	Электронный обмен данными	Обмен данными между различными базами данных	ГОСТ 6.20.1-90
OPC UA	OPC Unified Architecture		Стандарт основанный на OPC. Основное назначение - организация взаимодействия разнородных систем в рамках всего предприятия	MESA WP
SCA	Service Component Architecture	Сервис-компонентная архитектура	Группа спецификаций, предназначенных для разработки приложений на базе сервис-ориентированной архитектуры (SOA)	MESA WP
OEM(s)	Original equipment manufacturer(s)	Изготовитель комплектного оборудования		MESA WP
CAD	Computer Aided Design		Программный пакет, предназначенный для автоматизированного проектирования, разработки и производства конечного продукта, а также оформления конструкторской и/или технологической документации	ГОСТ Р 50.1.031-2001
OAGi	Open Applications Group, Inc		Ассоциация, объединяющая группу не коммерческих организаций по разработке стандартов (SDO)	www.oagi.org
CM(s)	Contract Manufacturer(s)	предприниматель-субподрядчик	организация, предоставляющая продукцию поставщику	ИСО 8402
	Collaborative Manufacturing(s)	«Объединенное производство» или «скоординированное производство»	Концепция управления сквозными процессами заказа, производства, поставок и потребления. Предполагает производство под заказ, единую информационную базу, высокую ориентированность производства на потребителя.	MESA WP
OASIS	Organization for the Advancement of Structured Information Standards		Организация, которая управляет разработкой, принятием промышленных стандартов электронной коммерции.	www.oasis-open.org
PSLX	Planning and Scheduling language on XML		Стандарт для языка XML при использовании его в системах производственного планирования	www.pslx.org

CRM	Customer Relationship Management	Система управления взаимодействием с клиентами		MESA WP
CAM	Computer Aided Manufacturing	технологическая подготовка производства	Имеет целью создание комплекта технологических документов: технологических маршрутов и операционных карт механообработки, сборки (монтажа), контроля; норм времени на выполнение технологических операций; управляющих программ для оборудования с числовым программным управлением; проектов оснастки и специального инструмента и т.д	ГОСТАНД АРТ РОССИИ P50.1.031-2001
BatchM L	Batch Markup Language	XML-реализация стандарта ISA-88		<a href="http://www.wbf.org/catalog/batchml.php">www.wbf.org/catalog/batchml.php</a>
APS	Advanced Planning and Scheduling	Детальное планирование		MESA WP
NPI	New Product Introduction	Внедрение нового продукта	Один из этапов жизненного цикла продукта	MESA WP
HMI (MMI)	Human-Machine Interface (Man-Machine Interface)	Интерфейс человеко-машинный		ГОСТ Р МЭК 60073-2000
SDO	Service Data Objects		Технология, предоставляющая унифицированную среду для разработки приложений обработки данных из различных источников и их унификации. Разработчик - сообщество открытых приложений Eclipse	<a href="http://www.eclipse.org/modeling/emf/?project=sdo">www.eclipse.org/modeling/emf/?project=sdo</a>
P/PE	Product and Process Engineering	Проектирование производственных процессов и продуктов		MESA WP
CSF(s)	Critical Success Factor(s)	Критические фактор(ы) успеха	Фактор, считающийся наиболее способствующим достижению успеха проекта	ISO 9001
SSM	Sales and Service Management	Управление сбытом и обслуживанием		MESA WP
BOM	Bill of Materials	состав изделия	Перечень компонентов изделия с указанием для каждого из них необходимых для его определения атрибутов (характеристик). Как правило, этот перечень представляет собой иерархически (древовидно) организованный список, отображающий вхождение деталей в подузлы, подузлов - в узлы и/или агрегаты и т.д.	ГОСТ Р 50.1.031-2001
BPR	Business-Process Reengineering	реинжиниринг бизнес-процессов	Комплекс мероприятий, направленных на совершенствование и повышение эффективности. Обычно включает в себя такие меры, как уточнение, объединение и/или разбиение функций, делегирование полномочий персонала «сверху вниз», устранение параллельных потоков и дублирования, выравнивание загрузки персонала и т.д.	ГОСТ Р 50.1.031-2001

OLE	Object Linking and Embedding		Программная технология, предоставляющая набор объектов, для обмена данными между приложениями	MESA WP
SQC	Statistical Quality Control	статистический контроль качества	Методы и виды деятельности оперативного характера, которые используют для выполнения требований к качеству, в которых применяют статистические методы	ГОСТ Р 50779.11-2000
NIIP/S MART	National Industrial Information Infrastructure Protocols Solutions for MES-Adaptable Replicable Technology			
IP	Intellectual Property/Internet Protocol	интеллектуальная собственность/интернет протокол		MESA WP
JIT	Just-In-Time	Точно вовремя	1. Технология организации производства без промежуточного складирования компонентов, которые поставляются точно в нужный момент времени 2. Стратегия принятия решений в самый последний подходящий для этого момент в целях обеспечения их максимальной точности	MESA WP
SOP(s)	Standard Operating Procedure(s)	Стандартная последовательность операций		ANSI/ISA95
MTO	Make-To-Order	Производство на заказ	Концепция, при которой производство инициируется внешним заказом.	ANSI/ISA95
CI	Continuous Improvement	Постоянное улучшение	Повторяющаяся деятельность по увеличению способности выполнить требования	ГОСТ Р ИСО 9000-2001
PDM	Product Data Management	Управление данными об изделии		ГОСТ Р 50.1.031-2001
Collab.	Collaborative (Collaborative Manufacturing)	см. Collaborative Manufacturing	Принятое сокращение	MESA WP
CAE	Computer Aided Engineering	автоматизированная разработка (и подготовка производства)	Система, предназначенная для концептуального проектирования на стадии эскизного проекта	ГОСТ Р 50.1.031-2001
SCP	Supply Chain Planning	Планирование цепочки поставок		MESA WP
RPC(s)	Remote Procedure Call(s)	удаленный вызов процедур	Технология, позволяющая приложениям вызывать функции или процедуры в другом адресном пространстве	MESA WP
WSDM	Web Services Distributed Management		Часть архитектура SOA. Определяет представлении интерфейсов управления произвольными ИТ-ресурсами	MESA WP WSDM OASIS Standard



GUI	Graphical User Interface	Графический интерфейс пользователя	Включает многооконный режим работы, полнографические редакторы, работу с мышью и т.д.	РД 153-34.2-35.520-99
DMZ	DeMilitarized Zone	Демилитаризованная зона	Часть компьютерной сети, находящаяся между локальной сетью и Интернетом. Обеспечивает выход в Интернет и внешнее присутствие в нём, предотвращая прямое обращение к локальной сети	MESA WP
BP	Best Practices	Передовой опыт (Успешные применения)		MESA WP
SCE	Supply Chain Execution	Управление цепочками поставок в реальном времени		MESA WP
COTS	Commercial Off The Shelf product	изделие неразрабатываемое	Имеющееся на рынке, доступное для приобретения изделие, обладающее требуемыми эксплуатационными характеристиками. Часто это коммерческое изделие, разработанное и используемое в других странах. Иногда такое изделие должно быть приспособлено, изменено или улучшено, чтобы выполнить заданные требования.	ГОСТ Р 50.1.031-2001
EPE	Equipment Procedural Element		Элемент модели ISA88. Описывает контролируемое оборудование: его свойства, состояние, возможные действия и т.д.	ANSI/ISA88
DRP	Distribution Resource Planning (sales, customer orders and distribution management)	управление продажами, заказами и дистрибуцией	Совокупность программных средств и данных, обеспечивающая выполнение следующих функций: sales, customer orders and distribution management; distribution resource planning; DRP - ведение списка клиентов (заказчиков); - формирование каталогов продукции и запасных частей; - формирование цен и прайс-листов; - формирование портфеля заказов и плана продаж; - подготовка и ведение договоров на поставку продукции; - подготовка и рассылка коммерческих предложений на поставку продукции; - расчет себестоимости заказов и договорных цен; - планирование сроков поставки продукции заказчику; - оценка статуса заказа относительно плана производства; - планирование потребностей и запасов в системе дистрибуции и др.	ГОСТ Р 50.1.031-2001
ETO	Engineer-To-Order	разработка на заказ	Концепция, при которой разработка инициируется внешним заказом.	ANSI/ISA95
CMMS	Computer Maintenance Management System	Система управления техническим обслуживанием	Программное обеспечение для диагностики и технического обслуживания оборудования	MESA WP

OSA	Open System Architecture	архитектура открытых систем	Международный стандарт	ISO/IEC 7498-1
DTD	Document Type Definition	Описание логической структуры данных	Файл, содержащий описание ИО и их атрибутов, а также правил, которым должна удовлетворять логическая структура БД.	ГОСТ Р 50.1.031-2001
TQM	Total Quality Management	Всёобщее (комплексное) управление качеством	Совокупность программных средств и данных, обеспечивающая выполнение функций, предписываемых международными стандартами ИСО 9000, в том числе: total quality management, TQM-сбор, хранение и статистическая обработка данных о входном контроле материалов и комплектующих;- сбор, хранение и статистическая обработка данных о результатах операционного контроля деталей и сборочных единиц (узлов, подузлов) в процессе производства;- сбор, хранение и статистическая обработка данных о результатах выходного контроля (приемосдаточных испытаний) готовых изделий;- формирование комплекта документов о качестве для конкретного экземпляра изделия (формуляра качества);- планирование, документирование и учет мероприятий по обеспечению качества в соответствии с ИСО 9000 и т.д. См. ГОСТ Р ИСО 9001, ГОСТ Р ИСО 9002, ГОСТ Р ИСО 9003, а также ИСО 8402	ГОСТ Р 50.1.031-2001 ИСО 8402
TOC	Theory of Constraints	Теория ограничений	Теория предлагает концентрировать организационные ресурсы на устранении ограничений, которые мешают компании полностью реализовать её потенциал. В России существует аналог - математический аппарат для точного производственного планирования на основе функции "производственной напряженности"	MESA WP Мауэрграуз Ю.Е.
ROA	Return on Assets	Рентабельность активов	выраженное в процентах отношение операционной прибыли к среднему за период размеру суммарных активов	MESA WP
Track.	Tracking		Принятое сокращение	MESA WP
Transform.	Transformation		Принятое сокращение	MESA WP
Visualiz.	Visualization		Принятое сокращение	MESA WP
Mgmt.	Management		Принятое сокращение	MESA WP
Mktg.	Marketing		Принятое сокращение	MESA WP
Optimiz.	Optimization		Принятое сокращение	MESA WP
Dem. Mgmt	Demand Management		Принятое сокращение	MESA WP
Cst.	Customization		Принятое сокращение	MESA WP
D.	Delivered		Принятое сокращение	MESA WP
Ops	Operations		Принятое сокращение	MESA WP
Mfg	Manufacturing		Принятое сокращение	MESA WP
Acq.	Acquisition		Принятое сокращение	MESA WP

CL	Collaborative		Принятое сокращение	MESA WP
TS	Transactional		Принятое сокращение	MESA WP
WG	Working Group	Рабочая группа	В ассоциации MESA действуют несколько выделенных направлений для работы, т. н. комитеты (рабочие группы), каждый из которых ведёт работы в определённой сфере профессиональной деятельности. Рабочие группы по системе показателей - Рабочая группа по стратегическим инициативам - Специальная группа по интересам: непрерывное производство - Комитет по решениям для нефтегазовой отрасли - Технический комитет - Комитет по организации веб-сайта - Комитет по маркетингу - Комитет по привлечению участников - Региональные рабочие группы	MESA WP

Глоссарий разработан при поддержке ООО «Ай-Си-Эс»

Спирин Константин Юрьевич, начальник отдела MES ООО «Ай-Си-Эс», член Российской рабочей группы MESA International

K.Spirin@mesarussia.ru

## СОДЕРЖАНИЕ

Предисловие .....	3
WP 27. Сервис-ориентированная архитектура в системах управления производством .....	5
Как ошибки оценки в ERP помогают распространению MES .....	61
Стандарты и технологии интеграции производственных информационных систем .....	68
Глоссарий терминологии MESA .....	85

## Информация о спонсорах:



Группа компаний «СМС-Автоматизация»  
443035, г. Самара, пр. Кирова 201, секция 9  
Тел./факс: +7 (846) 933-03-50  
[www.sms-automation.ru](http://www.sms-automation.ru)

Группа компаний «СМС-Автоматизация» основана в 1991г. Основные виды деятельности - создание и поддержка систем промышленной автоматизации (АСУ ТП) и дистрибуция техники автоматизации Siemens. В группе компаний работает 185 человек. Штаб-квартира группы и основные ресурсы находятся в Самаре. Офисы и производственные площадки расположены также в Москве, Новокуйбышевске, Елабуге, Чапаевске и Балаково и занимают более 2000 кв.м.

За время работы компании внедрено более 100 систем автоматизации технологических процессов в нефтегазовой, химической, энергетической и других отраслях промышленности. Основные клиенты компании - крупнейшие промышленные предприятия России, среди которых: РусГидро (Жигулевская ГЭС, Камская ГЭС), Роснефть (Новокуйбышевский НПЗ, Сызранский НПЗ, Куйбышевский НПЗ), Газпром (Уренгойгазпром), Волжская ТГК, КуйбышевАзот, Системный оператор Единой энергетической системы, МеталлоИнвест (Оскольский ЭМК, Лебединский ГОК).



ООО «Компания «ТЕРСИС»  
адрес: Россия, г. Москва, 109082, ул. Солянка 1/2, стр. 1  
тел./факс: +7 (495) 980-73-57  
[www.tersys.ru](http://www.tersys.ru)

Компания была основана в 1997 году и одной из первых в нашей стране выбрала приоритетным направлением деятельности область системной интеграции. Для своих партнеров компания оказывает полный спектр услуг: инжиниринг, поставка, монтаж, аутсорсинг, консультационная поддержка и необходимое обучение персонала. Такой подход наиболее полно удовлетворяет потребностям наших Заказчиков: они не тратят время на непрофильные задачи и получают в готовом виде работающие интегрированные решения для повышения эффективности своего бизнеса.

Деловая репутация и доверие заслуживается годами достойной работы, а теряется за один день. С первых дней работы в компании поддерживается понятие бесценности деловой репутации. Мы гордимся, что одинаково ответственно и внимательно подходим к решению задач всех своих Заказчиков: и крупных компаний, и компаний сектора СМБ, и частных лиц - это одно из принципиальных отличий нашего подхода к работе.



ООО «АМастер»  
адрес: Россия, г. Саратов, 410044, пр-кт Строителей, д. 1  
тел.: +7 (8452) 44-70-57  
тел./факс: +7 (8452) 44-70-70  
e-mail: [amaster@mail.saratov.ru](mailto:amaster@mail.saratov.ru)

ООО «АМастер» работает в сфере промышленной автоматизации более пяти лет. В основе технической политики фирмы лежит использование современных зарубежных технологий и лучших традиций отечественного инжиниринга. Компания занимается разработкой и сервисным обслуживанием автоматизированных систем управления технологическим процессом, систем ЧПУ, модернизацией действующего оборудования, а также поставкой средств автоматизации. В последние годы, в связи с интересом наших клиентов к MES-решениям, разработка и внедрение подобных систем стали одним из основных направлений деятельности компании.

Среди наших клиентов - производители строительных материалов, пищевой продукции, полиграфических изделий, бытовой техники, автомобильных комплектующих. География деятельности компании - не только Саратов и Саратовская область, но и Пензенская, Ульяновская, Волгоградская области и другие регионы Приволжского федерального округа.

**MES - теория и практика.** Выпуск 2 (2010). Официальные материалы ассоциации MESA International. Москва, 2010.-80 с, илл.

Составители: И.С. Решетников, А.П. Козлецов

© Российская рабочая группа MESA International, 2010

# ИНФОКОНТ - платформа для построения информационной системы производства

## Информационная система производства

- ✓ единая точка доступа ко всем данным для людей и систем
- ✓ интеграция данных по производству (АСУТП, ТМ), энергоресурсам (АСУТГ, АСКУЭ) и др.
- ✓ отображение на сводных формах и детальных экранах
- ✓ совмещение плана, задания и факта

## Просто попробуйте!

- ✓ посетите демосайт <http://infocont.ru> и поработайте с информационной системой производства на базе Инфоконт
- ✓ закажите **бесплатное пилотное внедрение** для вашего производства\*

\* Условия и срок проведения акции - на сайте <http://infocont.ru>



ГРУППА КОМПАНИЙ



АВТОМАТИЗАЦИЯ

[www.sms-automation.ru](http://www.sms-automation.ru)