

**Сервис-  
ориентированные  
архитектуры  
SOA**

# Что такое SOA












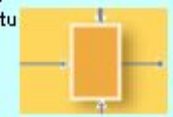






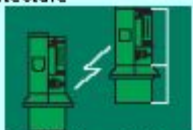








- SOA (Service-Oriented Architecture) – средство реализации гибких архитектурных решений для развивающихся КИС на базе слабосвязанных сервисов;
- Сервис – многократно используемый масштабируемый программный компонент КИС, обеспечивающий выполнение предписанных функций обработки данных и обмена информацией с внешней средой посредством асинхронного обмена сообщениями.

# Понятие «Корпоративная архитектура»

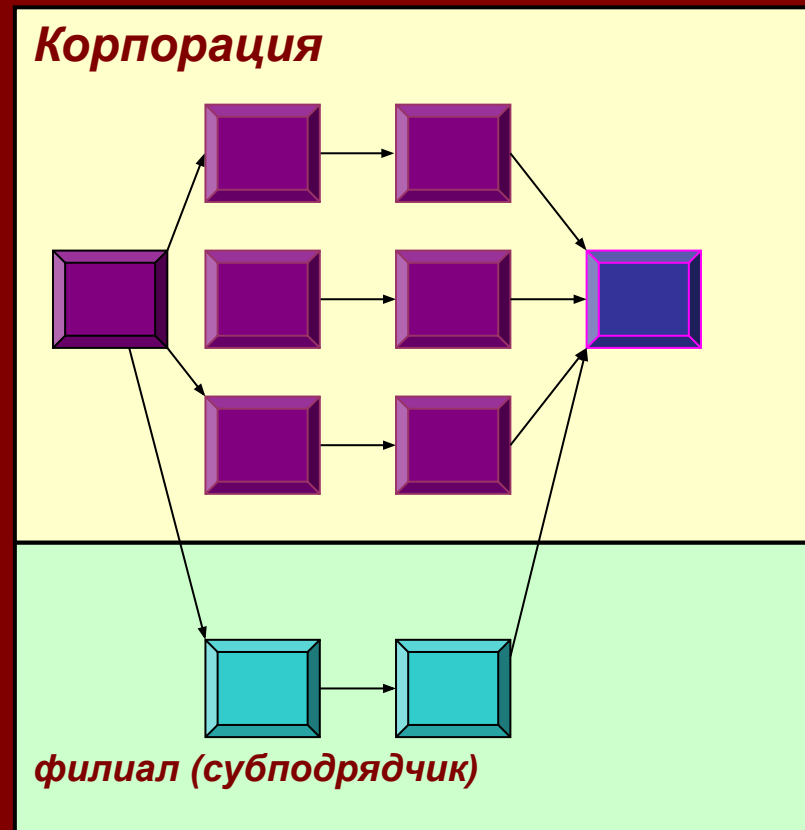
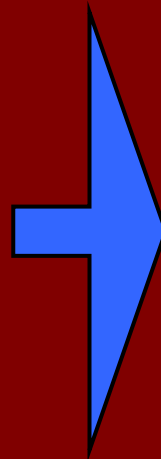
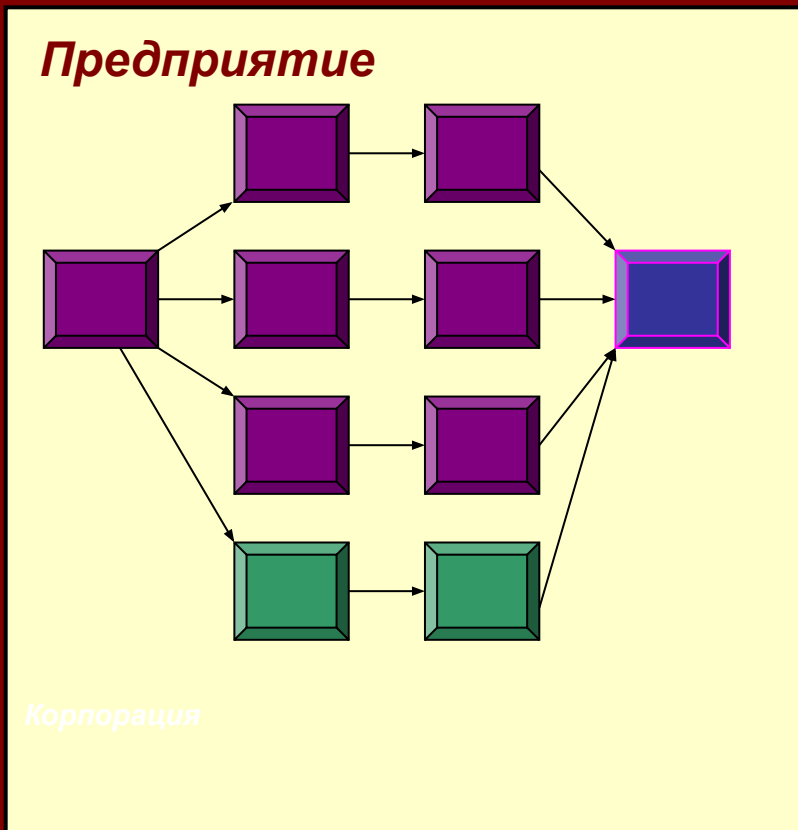
- «Корпоративная архитектура предприятия – это структурированное описание делопроизводства и бизнес-процессов предприятия, приложений и методов автоматизации, поддерживающих бизнес-процессы, а также информация, технологии и инфраструктура, необходимые для их выполнения. Архитектура позволяет выработать целостный план работ и скоординированных проектов, необходимых для претворения в жизнь задач развития ИТ-инфраструктуры предприятия»

([www.technical-translation.com](http://www.technical-translation.com))

# Корпоративная архитектура – основа SOA

	ДАННЫЕ	ФУНКЦИИ	СЕТЬ	ЛЮДИ	ВРЕМЯ	ЦЕЛИ
<b>МИССИЯ И СТРАТЕГИЯ ПРЕДПРИЯТИЯ</b>	List of Things Important to the Business  Entity=Class of Business Thing	List of Processes the Business Performs  Process=Class of Business Process	List of Locations in Which the Business Operates  Node=Major Business Location	List of Organizations Important to the Business  People=Major Organizational Unit	List of Events/Cycles Significant to the Business  Time=Major Business Event/Cycle	List of Business Goals/Strategies  Ends/Means=Major Business Goal/Strategy
<b>Аналитики, топ-менеджеры</b>						
<b>КОНЦЕПТУАЛЬНАЯ БИЗНЕС-МОДЕЛЬ</b>	e.g., Semantic Model  Entity=Business Entity Relationship=Business Relationship	e.g., Business Process Model  Process=Business Process I/O=Business Resources	e.g., Business Logistics of System  Node=Business Location Link=Business Linkage	e.g., Work Flow Model  People=Organizational Work=Work Product	e.g., Master Schedule  Time=Business Event Cycle= Business Cycle	e.g., Business Plan  End=Business Objective Means=Business Strategy
<b>Инвесторы, акционеры</b>						
<b>ЛОГИЧЕСКАЯ МОДЕЛЬ СИСТЕМЫ</b>	e.g., Logical Data Model  Entity=Data Entity Relationship=Data Relationship	e.g., Application Architectu  Process=Application Function I/O=User Views	e.g., Distributed System Architecture  Node=I/S Function (Processor, Storage, e) Link=Line Characteristic	e.g., Human Interface Architecture  People=Role Work=Delliverable	e.g., Processing Structure  Time=System Event Cycle=Processing Cycle	e.g., Business Rule Mo  End=Structural Assertion Means =Action Assertion
<b>Системные архитекторы</b>						
<b>ТЕХНОЛОГИЧЕСКАЯ МОДЕЛЬ</b>	e.g., Phisical Data Model  Entity=Segment/Table/etc Relationship=Pointer/Key/...	e.g., System Design  Process=Computer Function I/O=Data Elements/Sets	e.g., Technology Architecture  Node=Hdw/Sys. Software Link=Line Specification	e.g., Presentation Architecture  People=User Work=Screen Formats	e.g., Control Structure  Time=Execute Cycle= Component Cycle	e.g., Rule Design  End=Condition Means=Action
<b>Разработчики</b>						
<b>ДЕТАЛЬНОЕ ПРЕДСТАВЛЕНИЕ</b>	e.g., Data Definition  Entity=Field Relationship=Address	e.g., Program  Process=Language Statement I/O=Control Block	e.g., Network Architecture  Node=Address Link=Protocol	e.g., Security Architecture  People=Identity Work=Job	e.g., Timing Definition  Time=Interrupt Cycle= Machine Cycle	e.g., Rule Specification  End=Sub-condition Means =Step
<b>Программисты</b>						
<b>ФУНКЦИОНИРУЮЩАЯ СИСТЕМА</b>	e.g.: DATA	e.g.: FUNCTION	e.g.: NETWORK	e.g.: ORGANIZATION	e.g.: SCHEDULE	e.g.: STRATEGY

# Компонентная модель архитектуры EA (SOE- Service Oriented Enterprise)



**EA – Enterprise Architecture**

# Бизнес-процессы и бизнес-компоненты – «материальные блоки» для реализации SOA

Бизнес-процессы (БП) представляют собой структурированное описание заданной последовательности выполняемых исполнителями элементарных технологических шагов (операций), состоящих в преобразовании имеющихся у предприятия ресурсов (материальных, информационных, кадровых, энергетических, финансовых и пр.), имея конечной целью выпуск готовых изделий или реализацию услуг согласно производственной программе (плану) по объему и номенклатуре.

Под бизнес-компонентами (БК) понимается законченная совокупность функциональных компонентов ЕА, поддерживаемых соответствующими средствами ИКТ-инфраструктуры предприятия (корпорации). БК представляет собой независимый самоуправляемый модуль, который может функционировать не только в любой оргструктуре корпорации, но даже вне ее. В такой трактовке БК представляют собой своеобразные крупные «строительные блоки» для проектирования и практической реализации заданной архитектурной бизнес-модели предприятия.



# Особенности построения ЕА и КИС на основе БК

Каждый БК представляет собой объект, способный воспринимать и обрабатывать входные материальные потоки и оперировать ими с целью получения необходимых результатов на выходе. БК могут включать в себя людей, ресурсы, технологии, сведения типа «know-how» и другие составляющие, необходимые для выполнения тем или иным компонентом предписанного множества бизнес-функций (бизнес-процессов).

Архитектура предприятия, в основу которой положены БК, позволяет достаточно легко осуществлять декомпозицию и детальное исследование существующей модели ведения бизнеса, выявлять ее недостатки и узкие места, оценивать критические факторы влияния со стороны внешней бизнес-среды (например, рыночной конъюнктуры), а затем вырабатывать обоснованные рекомендации по усовершенствованию ЕА на основе новой композиции БК, или, в более серьезных случаях, при помощи коренной их реконструкции. Тем самым обеспечивается одно из основных требований времени - высокая реактивность и адаптивность бизнеса (business agility).

Дальнейшее развитие идей компонентной организации бизнес-модели предприятия приводит к концепции сервисно-ориентированного предприятия (SOE – Service-Oriented Enterprise).

# SOE - развитие компонентной модели бизнеса

Согласно представлениям о функционировании SOE, сервисы представляют собой средства формирования законченных материальных продуктов (изделий или услуг), которые поставляются в бизнес-среду (БС) как товар. Сервисы также имеют возможность взаимодействовать с окружающей БС и другими сервисами посредством некоторых стандартных интерфейсов (сообщений).

Таким образом, модель функционирования SOE как одного из перспективных вариантов построения ЕА корпоративного уровня в рассмотренной трактовке представляется как множество взаимодействующих друг с другом простых или составных (сложных) сервисов, которые обмениваются материальными потоками и/или информацией в процессе обработки событий. Правила взаимодействия бизнес-компонентов в процессе предоставления услуг посредством сервисов определяются соответствующими формальными соглашениями между ними (контрактами), оговаривающими стоимость, количество и качество предоставляемых сервисами услуг.

Рассмотренные инновационные сдвиги в практических сферах понимания законов функционирования современных предприятий, базирующихся на корпоративной бизнес-модели, с неизбежностью приводят к смене базовой парадигмы архитектурной реализации КИС.



# Классическая задача SOA

Основная задача SOA заключается в создании архитектурной ИТ-модели бизнеса компании, которая обеспечивает быструю сборку слабо связанных распределенных программных объектов этой модели в единую среду исполнения. Сервисы в рамках SOA являются средствами реализации этих распределенных программных компонентов.

Таким образом, сервис можно определить как многократно используемый, масштабируемый компонент КИС, который обеспечивает выполнение предписанных ему функций обработки данных и обменивается информацией с внешней средой посредством асинхронного обмена сообщениями.

# ПАРАДИГМА SOA

- Реализация распределенных программных компонентов в виде слабосвязанных сервисов;
- Набор унифицированных спецификаций для описания и организации взаимодействия сервисов;
- Композиция простых сервисов в любые более сложные структуры («оркестровка» сервисов);
- Быстрая сборка распределенных программных объектов в единую среду исполнения (сервисная «хореография»);
- Возможность легкой миграции сервисов по гетерогенным сервисным платформам.

# Основные преимущества SOA

- Возможность построения гибких адаптируемых архитектур, обеспечивающих оперативные функциональные перестройки и постоянное развитие на протяжении жизненного цикла КИС;
- Быстрота реализации целевой архитектуры КИС;
- Легкость и простота масштабирования сервис-ориентированных приложений;
- Мобильность и интероперабельность сервисов;
- Легкость внесения функциональных расширений, модернизаций и дополнений в виде новых слабосвязанных сервисов;
- Простота удаления морально устаревших сервисных компонентов.

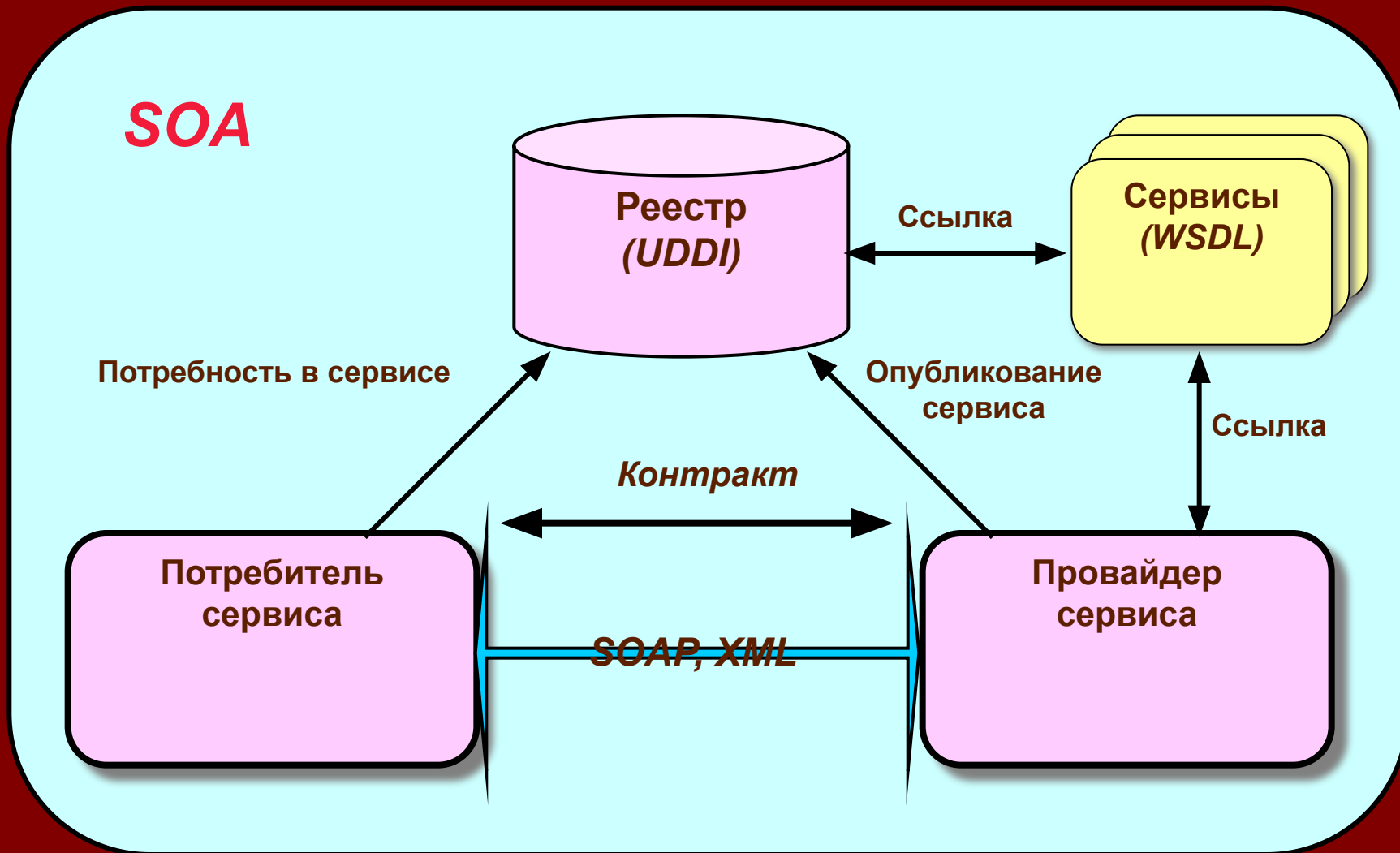
# Базовые составляющие SOA

- WSDL (Web Services Description Language) – язык описания сервисных интерфейсов;
- UDDI (Universal Description, Discovery and Integration) – стандарт описания программных интерфейсов реестра сервисов;
- SOAP (Simple Object Access Protocol) – стандарт описания методов взаимодействия объектов в SOA.

# Краткая характеристика компонентов SOA

- **WSDL** – средство подготовки контрактов на предоставление сервисов (спецификация ожидания) в нотации XML;
- **UDDI** – спецификация (интерфейсов) реестра сервисов, предоставляющая общую для различных инструментальных платформ базу для взаимно совместимых технологий описания, публикации, обнаружения и вызова сервисов;
- **SOAP** – предоставляет возможности по организации выполнения сервисов в процессе их реинкарнации и вызова для исполнения.

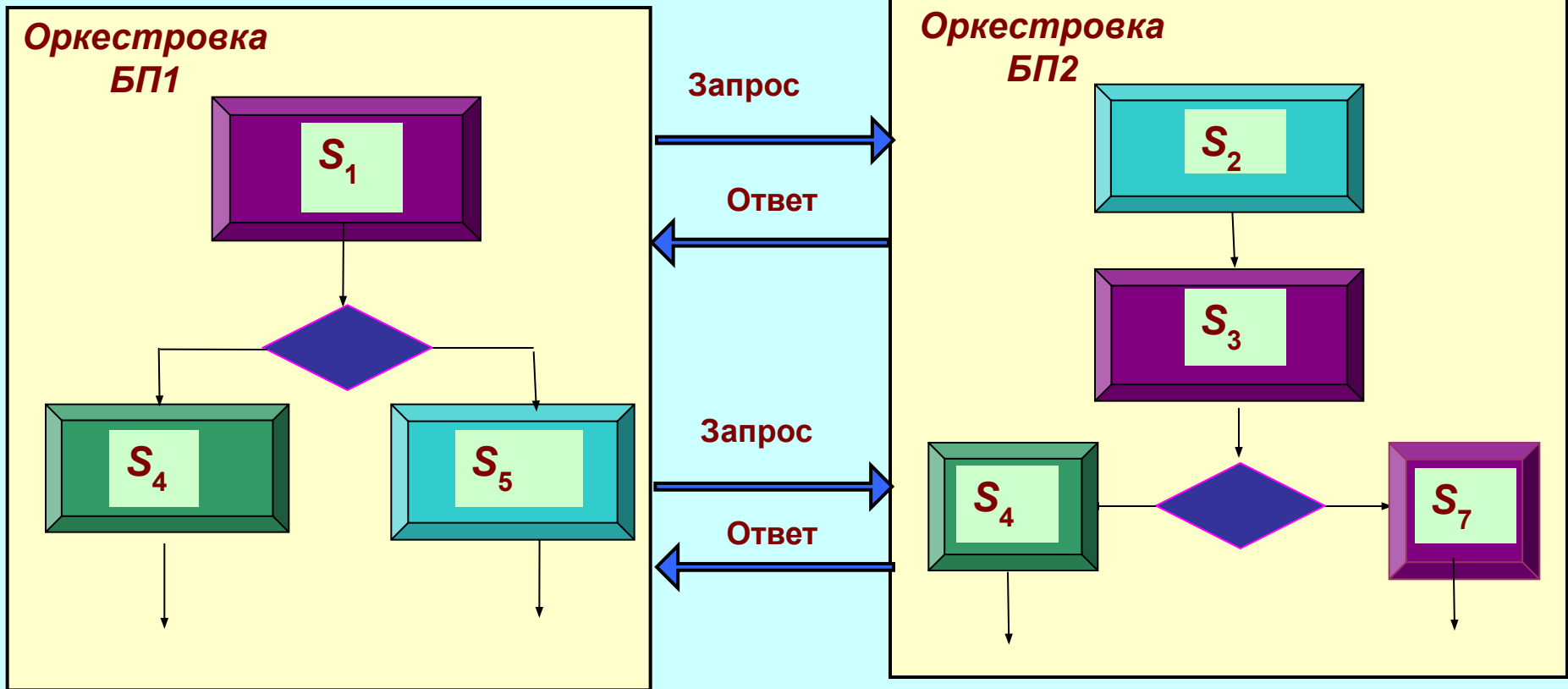
# Классическая схема представления SOA



# «Оркестровка» и «хореография» сервисов

SOA

Хореография





# Что такое «оркестровка»

В рамках компонентной модели ЕА бизнес-компоненты выполняют свои функции, обращаясь в определенной последовательности к независимым сервисам, реализуя в конечном итоге формальную бизнес-модель предприятия. Компоновка действий, выполняемых при этом отдельными приложениями (базовыми сервисами) в целостный бизнес-процесс, называется «оркестровкой».

В отличие от ранее принятых подходов к интеграции сложных ООП-приложений, в процессе «оркестровки» важна именно логическая последовательность вызова сервисов, а не особенности конкретных конфигураций ИТ-платформ, необходимых для реализации механизмов взаимодействия сервисов. Поэтому механизм «оркестровки» обеспечивает необходимую гибкость и масштабируемость при выполнении БП, обрабатывая потоки работ и определяя дальнейшую последовательность действий.

Возможна также компоновка высокоуровневых сложных сервисов из существующих «оркестрованных» бизнес-процессов (так называемая рекурсивная композиция).

# Понятие «хореографии»

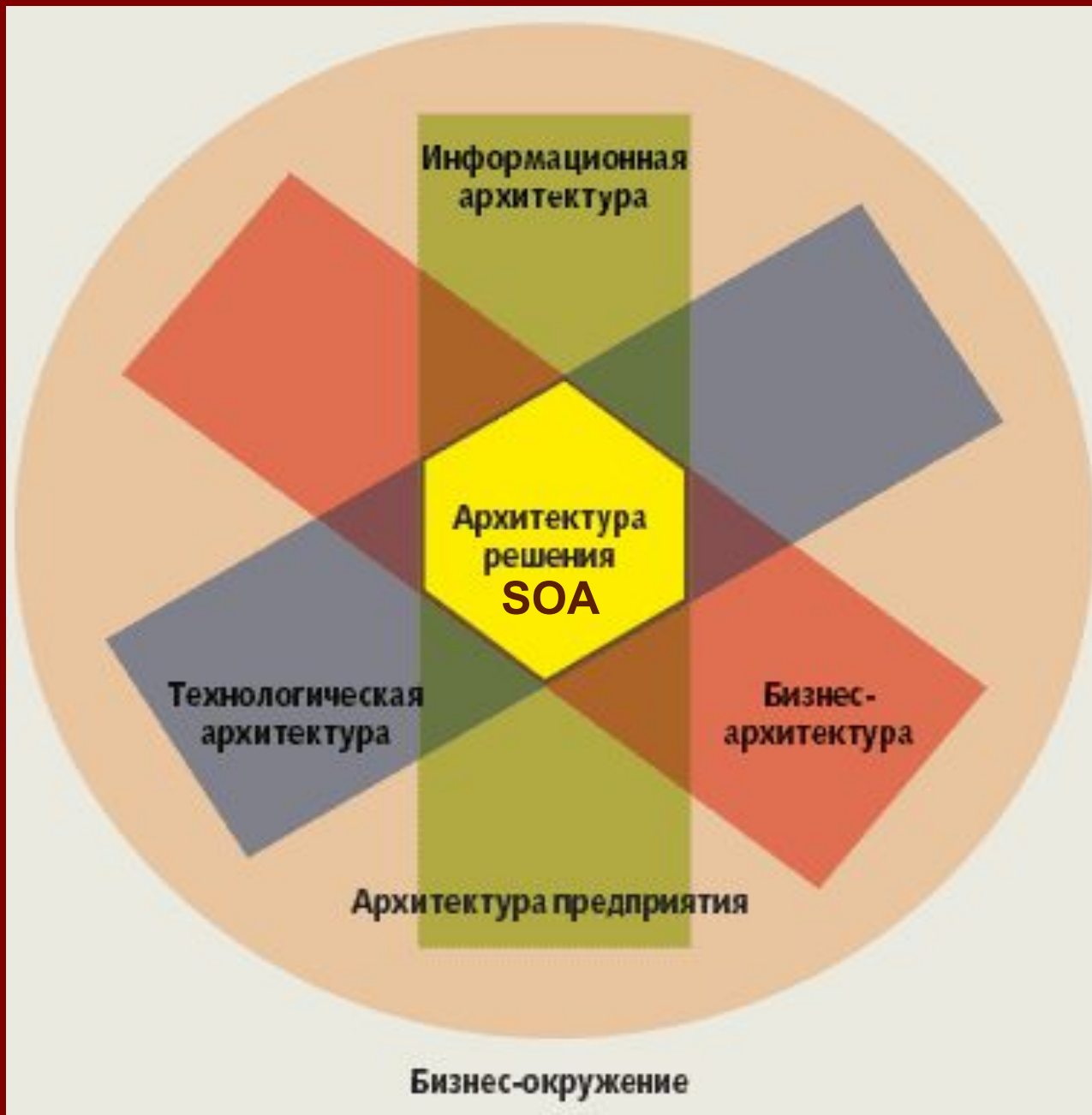
Процесс, называемый «хореографией», отражает правила (протоколы) взаимодействия участников, каждый из которых выполняет свои бизнес-функции в сложных БП корпоративного масштаба, на уровне внешних (публичных) обменов сообщениями. Таким образом, «хореография» решает проблему асинхронного установления связей между исполняемыми процессами. «Хореография» не является исполняемым процессом и относится к классу кодексов, обеспечивающих соблюдение необходимых протольно-процедурных формальностей при установлении отношений между взаимодействующими сторонами.

# Взаимосвязь понятий «оркестровки» и «хореографии»

**«Оркестровка» описывает полностью автономный БП, т. е. частный поток работ, контролируемый одним субъектом (БК) корпоративного бизнеса. Она позволяет организовать необходимую последовательность и логику распределения работ между сервисами в ходе решения конкретной бизнес-задачи.**

**«Хореография» отвечает за конкретную организацию взаимодействия субъектов – она описывает правила (протоколы) их обращений друг к другу. Это – модель политик, или модель описания бизнес-правил, которые являются наиболее часто и динамично изменяемыми элементами в течение жизненного цикла (ЖЦ) в рамках корпоративной бизнес-модели.**

# Модель архитектуры предприятия (GEAF)



**GEAF- Gartner  
Enterprise  
Architecture  
Framework**

# Технологическая цепочка создания КИС в рамках парадигмы SOA

- Создание высокоуровневой бизнес-модели компании;
- Проектирование бизнес-процессов (БП) на основе БМ с применением средств и методов BPM (Business Process Management);
- Моделирование, анализ и оптимизация БП;
- Описание и формирование сервисов SOA как средств реализации БП с применением языков типа BPML, BPEL (Business Process Modeling/Execution Language) – преобразование моделей в программную архитектуру;
- Формирование потоков процессов и машин состояний;
- Создание сложных (составных) сервисов SOA и их логическая компоновка в исполняемые модели БП на основе «оркестровки» - сборка сервисов;
- Задание потоков управления, политик и бизнес-правил;
- Тестирование и внедрение;
- Исполнение и мониторинг БП, включая асинхронные аспекты взаимодействия компонентов («хореография»)

# Общая архитектура SOA

(Концептуальная модель IBM SOA Foundation)

Бизнес-  
процессы

Язык исполнения бизнес-процессов  
для сервисов (BPEL)

Качество  
обслуживания  
(QoS)

Надежность

Транзакции

Управление

Безопасность

Описание

Язык описания интерфейсов сервисов (WSDL)

Обмен  
сообщениями

SOAP

Расширенный язык разметки (XML)

Другие протоколы,  
другие сервисы

# Пятиуровневая модель зрелости SOA





# Распределение сервисов по уровням



# Этапы адаптации модели ЕА к реализации в рамках SOA

- Реализация программных модулей системы в виде отдельных сервисов;
- Сервис-ориентированная интеграция бизнес-функций, включая заключение «унаследованных» приложений в сервисные «оболочки» и их дальнейшее представление в виде полноценных сервисов;
- Трансформация всей ИКТ-инфраструктуры в корпоративном масштабе (виртуализация всех ИТ-ресурсов);
- Изменения в корпоративной бизнес-модели - переход к новой архитектуре ЕА, построенной на базе бизнес-компонентов.

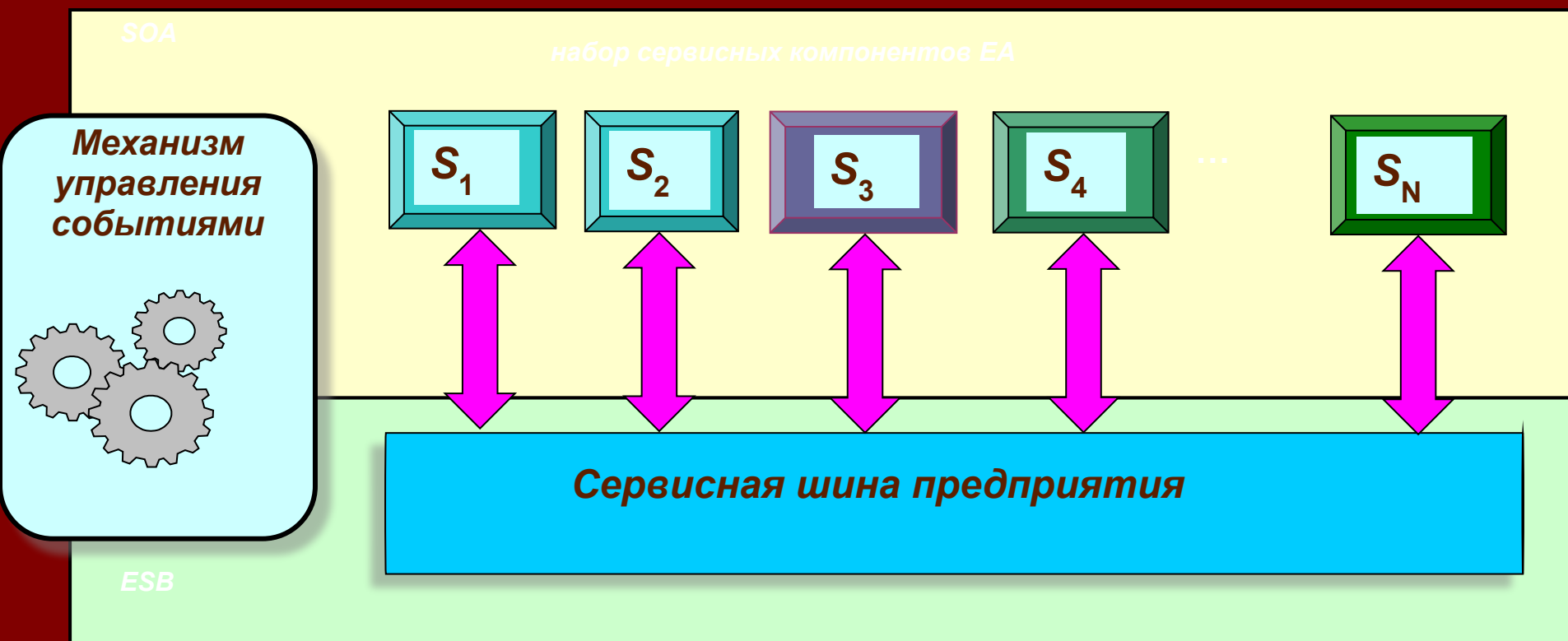
# Расширение SOA на принципах EDA/ESB

Способы и механизмы взаимодействия сервисов во времени в процессе реализации корпоративной бизнес-модели, методы и средства доставки результатов их исполнения конечным пользователям, некоторые другие механизмы взаимодействия сервисов (маршрутизация, подписка/ рассылка сообщений и др.) априорно лежат вне базовой модели SOA.

Для отражения этих моментов служат другие модели, тесно связанные с SOA, например, модели архитектуры, управляемой событиями, (EDA – Event-Driven Architecture) и сервисной шины предприятия (ESB – Enterprise Service Bus).

Событийная модель EDA, основанная на привязке EA к реально происходящим в бизнес-среде изменениям (поток внешних событий), объективно отражает событийную природу окружающего мира. Взаимодействие сервисов в процессе обработки происходящих в бизнес-среде событий, реализующее поддержку функционирования SOA в соответствии с моделью EDA, осуществляется через сервисную шину предприятия ESB.

# Сервисная шина предприятия (ESB)



**ESB – Enterprise Service Bus**

# Основные функции и задачи ESB

Сервисная шина ESB предназначена для практической реализации механизмов управления взаимодействием сервисов и маршрутизацией сообщений. Она служит универсальной средой для приема, обработки и передачи запросов на сервисы, а также для рассылки результатов их работы заинтересованным в этих результатах потребителям.

Наличие шины ESB обеспечивает единый стандартный интерфейс для организации взаимодействия во времени и пространстве всех видов корпоративных сервисов, включая унаследованные и вновь развертываемые.

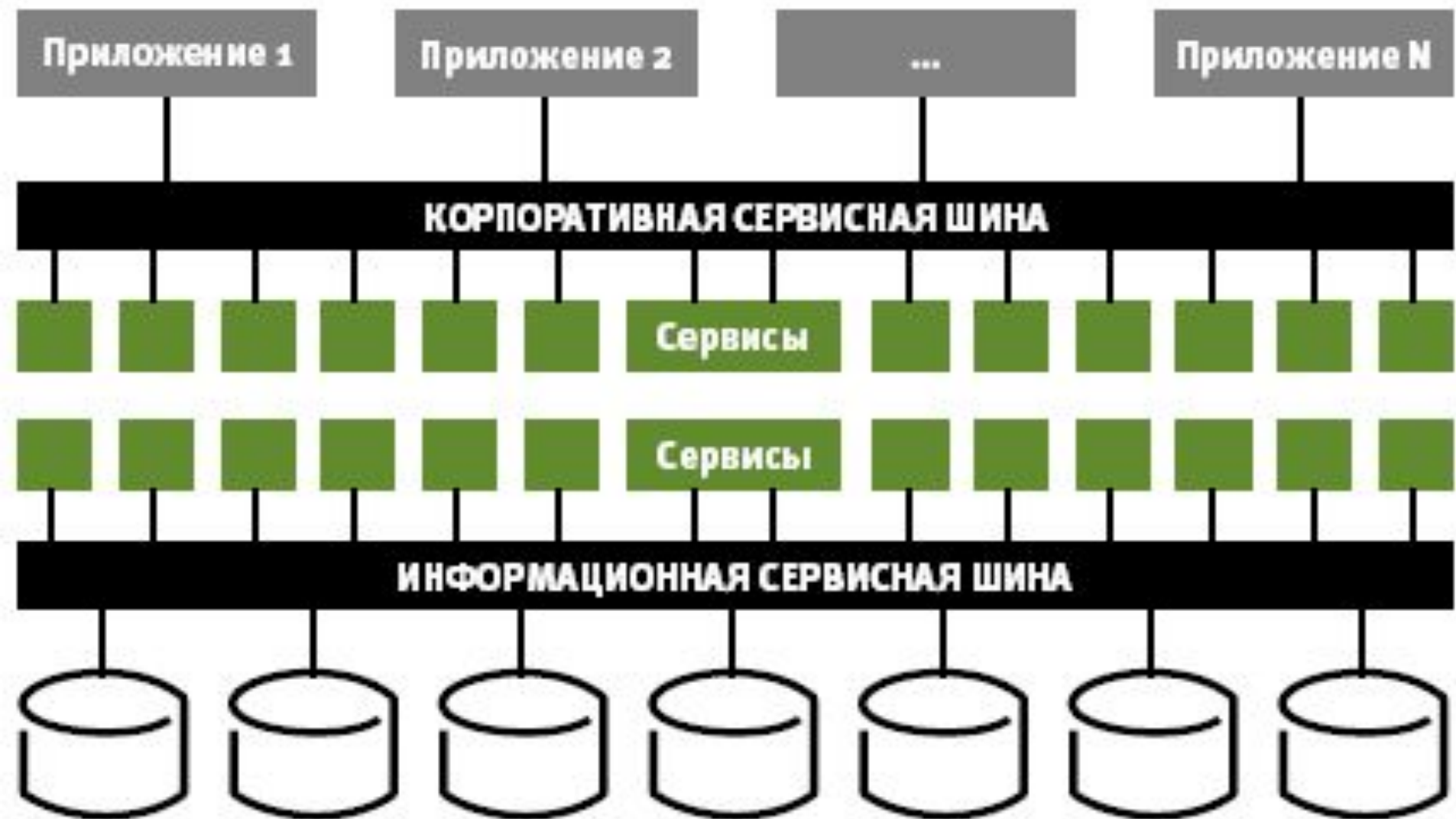
Следует подчеркнуть, что ESB – это не физическая инфокоммуникационная среда, а логическая модель для описания процессов взаимодействия сервисов в условиях развертывания потоков событий в реальной бизнес-среде в реальном масштабе времени.

С другой стороны, это достаточно конкретная технология реализации EDA и SOA, опирающаяся на виртуализацию ИКТ-ресурсов как на основу функционирования сервисной модели.

# Типовые функциональные модули ESB

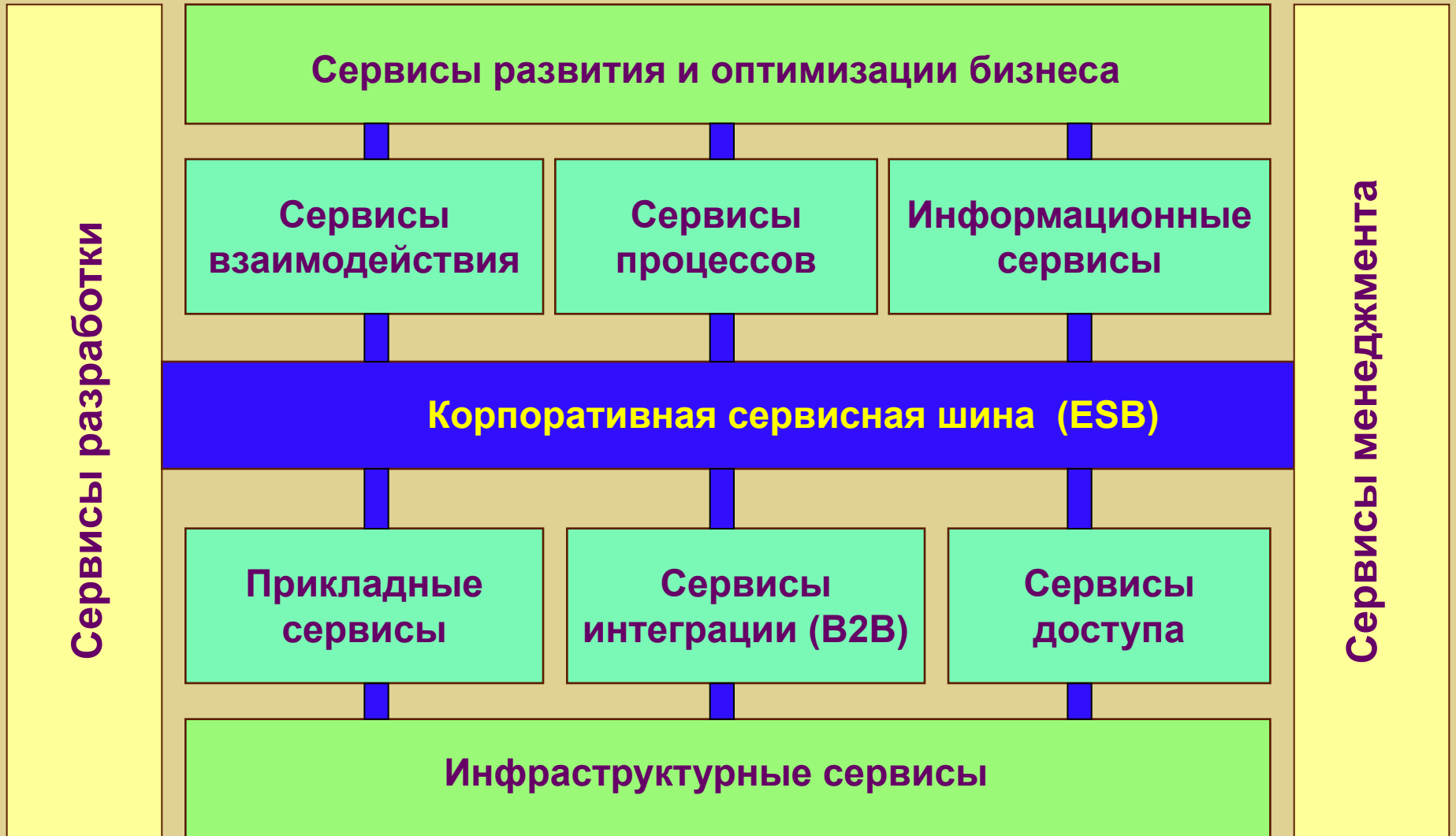
- транспортировка событий;
- шина распространения событий;
- обеспечение жизнеспособности событий;
- подписка на события;
- обновление событий;
- обработка очередей событий;
- база событий и метаданных событий.

# Сервисная и информационная шины (ESB + ISB)





# Логическая модель SOA



# Основные сервисы логической модели SOA

- Сервисы взаимодействия – обеспечивают связь между приложениями и внешними источниками данных;
- Сервисы процессов (СП) - реализуют логику управления, в т.ч. организуют потоки БП и управляют машинами состояний, осуществляют «оркестровку» процессов;
- Информационные сервисы – выполняют обеспечение БП данными, осуществляют миграцию данных и наполнение хранилищ данных, обеспечивают поддержку бизнес-аналитики, управляют ЖЦ информации;
- Прикладные сервисы – реализуют ядро бизнес-логики, связываются в единые БП с помощью СП;
- Сервисы интеграции – ориентированы на организацию работы с приложениями и данными бизнес-партнеров;
- Сервисы доступа – обеспечивают включение в SOA «унаследованных» приложений путем их инкапсуляции.

# Вспомогательные сервисы модели SOA

- Сервисы развития и оптимизации бизнеса – включают инструментарию и структуры описания метаданных, средства для моделирования БП, для измерения метрик процессов и анализа производительности;
- Сервисы разработки – состоят из инструментариев моделирования архитектуры, средств разработки и сборки SOA-приложений, средств их отладки, а также механизмов публикации, обнаружения и вызова сервисов (на базе интерфейсов);
- Сервисы менеджмента – служат для мониторинга работы всех сервисов SOA, анализа сбоев и «узких» мест, восстановления после сбоев и реализации установленной административной политики;
- Инфраструктурные сервисы – образуют ядро среды исполнения сервисов SOA, при необходимости осуществляют виртуализацию компьютерных платформ.

# Физическая многослойная модель SOA

1	Транспортный слой (Transport layer)	Описывает средства обмена данными между веб-сервисами	HTTP, JMS, SMTP	WS-ReliableMessaging, BEEP
2	Коммуникационный слой (Service communication layer)	Описывает средства формализации механизмов использования транспортных протоколов веб-сервисами.	SOAP	REST
3	Слой описаний сервисов (Service description layer)	Описывает средства формализации интерфейсов веб-сервисов с целью обеспечения их функционирования независимо от программно-аппаратной платформы реализации или языка программирования. Различают два вида описаний сервиса: 1.операционное (operational); 2.полное (complete)	XML, WSDL	ebXML
4	Сервисный слой (Service layer)	Описывает программное обеспечение, вызываемое с помощью WSDL-описаний интерфейсов веб-сервисов. В частности, это сами веб-сервисы		
5	Слой бизнес-процессов (Business process layer)	Описывает возможности организации веб-сервисов для реализации бизнес-процессов и потоков работ. При этом определяются правила, задающие последовательность взаимодействия веб-сервисов с целью удовлетворения бизнес-требованиям	в настоящее время нет	BPEL4WS
6	Слой реестров сервисов (Service registry layer)	Описывает возможности организации веб-сервисов в иерархические библиотеки, позволяющие публикацию, поиск и вызов веб-сервисов по их WSDL-описаниям интерфейсов	UDDI	WS-Inspection
7	Слой политик (Policy layer)	Описывает правила и условия, согласно которым веб-сервисы могут быть использованы. Поскольку данные правила и условия относятся как к функциональному аспекту веб-сервисов, так и к аспекту обеспечения качества сервиса на Рис. 1, данный слой является общим для обоих аспектов	в настоящее время нет	WS-Policy, WS-PolicyAssertions и WS-PolicyAttachment
8	Слой безопасности (Security layer)	Описывает возможности обеспечения безопасности веб-сервисов и безопасности их функционирования (авторизация, аутентификация и разделение доступа)	WS-Security	WS-SecureConversation, WS-Federation, WS-Trust, WS-Authorization, и WS-Privacy
9	Слой транзакций (Transaction layer)	Описывает свойство транзакционности распределенных систем на основе веб-сервисов для обеспечения надежности их функционирования	в настоящее время нет	WS-Transaction и WS-Coordination
10	Слой управления (Management layer)	Описывает возможности управления веб-сервисами и характеристиками их функционирования		

# Стек технологий Web-сервисов

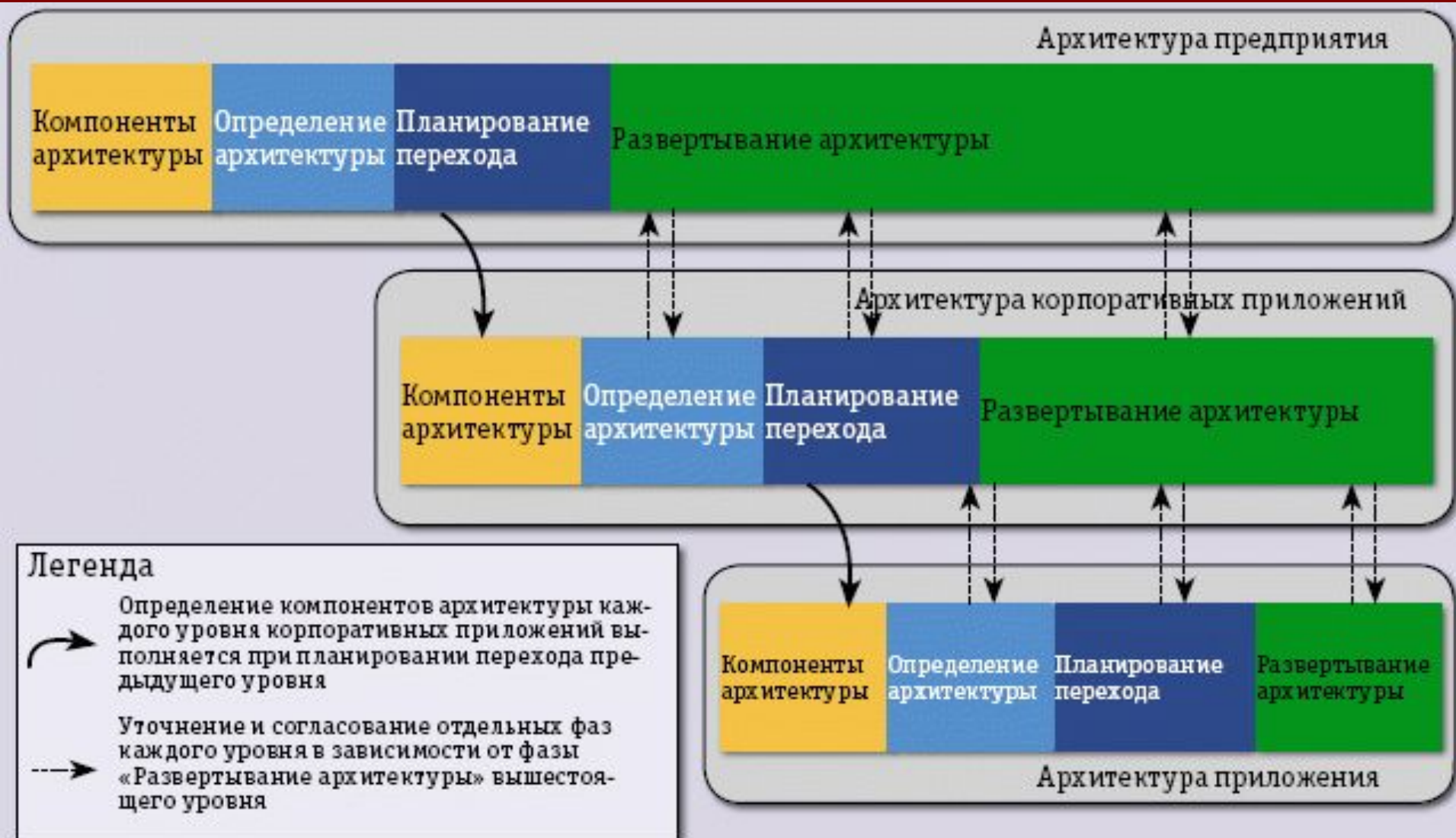
## Функциональность

Слой реестров сервисов	WS-Inspection	Слой бизнес-процессов	
			BPEL4WS
UDDI		Сервисный слой	
		Слой описаний сервисов	
		XML, WSDL	ebXML
		Коммуникационный слой	
		SOAP	REST
		Транспортный слой	
		HTTP, JMS, SMTP	WS-ReliableMessaging, BEEP

## Качество сервиса

Слой политик		WS-Policy, WS-PolicyAssertions и WS- PolicyAttachment	
Слой безопасности	WS-Security	WS-SecureConversation, WS- Federation, WS-Trust, WS- Authorization, и WS-Privacy	
Слой транзакций		WS-Transaction и WS-Coordination	
Слой управления			

# Модель архитектуры для реализации SOA





# Взаимосвязь архитектуры, IT-стратегии и SOA





# Взаимосвязь архитектурных моделей ЕА и SOA



Модель eTOM – корпоративная бизнес-модель для отрасли связи

# Взаимосвязь EA, BPM и SOA

- Бизнес-процессы могут быть представлены средствами BPM (Business Process Management), отлажены путем моделирования, а затем реализованы средствами SOA в виде соответствующих сервисов, являющихся «строительными блоками» архитектуры предприятия;
- Таким образом, SOA из инструментария архитектора ИТ-систем, рассматривавшегося ранее только как средство реализации чисто «программистской» технологии взаимодействия слабосвязанных модулей, быстро превращается в инструмент реализации всей глобальной корпоративной архитектуры EA.

# Модель интеграции BPM и SOA



# Организация взаимодействия BPM и SOA

eTOM

*Бизнес-модель компании*

**BPM**

Моделирование

Анализ

Реализация

Мониторинг

**SOA**

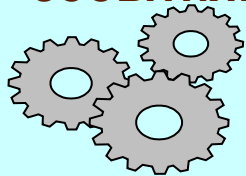
Прикладные  
сервисы

Информационные  
сервисы

Сервисы  
процессов

Сервисы  
интеграции (B2B)

Механизм  
управления  
событиями



*Сервисная шина предприятия (ESB)*

Сервисы  
мониторинга

Служебные  
сервисы

Технические  
сервисы

Сервисы доступа к  
унаследованным  
приложениям

Сервисы доступа к  
данным

Сервисы доступа к  
сетевым ресурсам

# Инструментальные средства для SOA

Основные артефакты и функции для поддержки конструкций SOA могут быть описаны на специальных языках. Одним из наиболее удобных и популярных из них является язык исполнения бизнес-процессов BPEL (Business Process Execution Language), который фактически является расширением WSDL. BPEL поддерживает как исполняемые, так и абстрактные процессы, определяя независимые от технологических платформ конструкции для реализации исполняемых БП и протоколы взаимодействия на уровне обмена сообщениями. Это обеспечивает возможность интеграции гетерогенных прикладных компонентов в единую интероперабельную и хорошо масштабируемую модель.

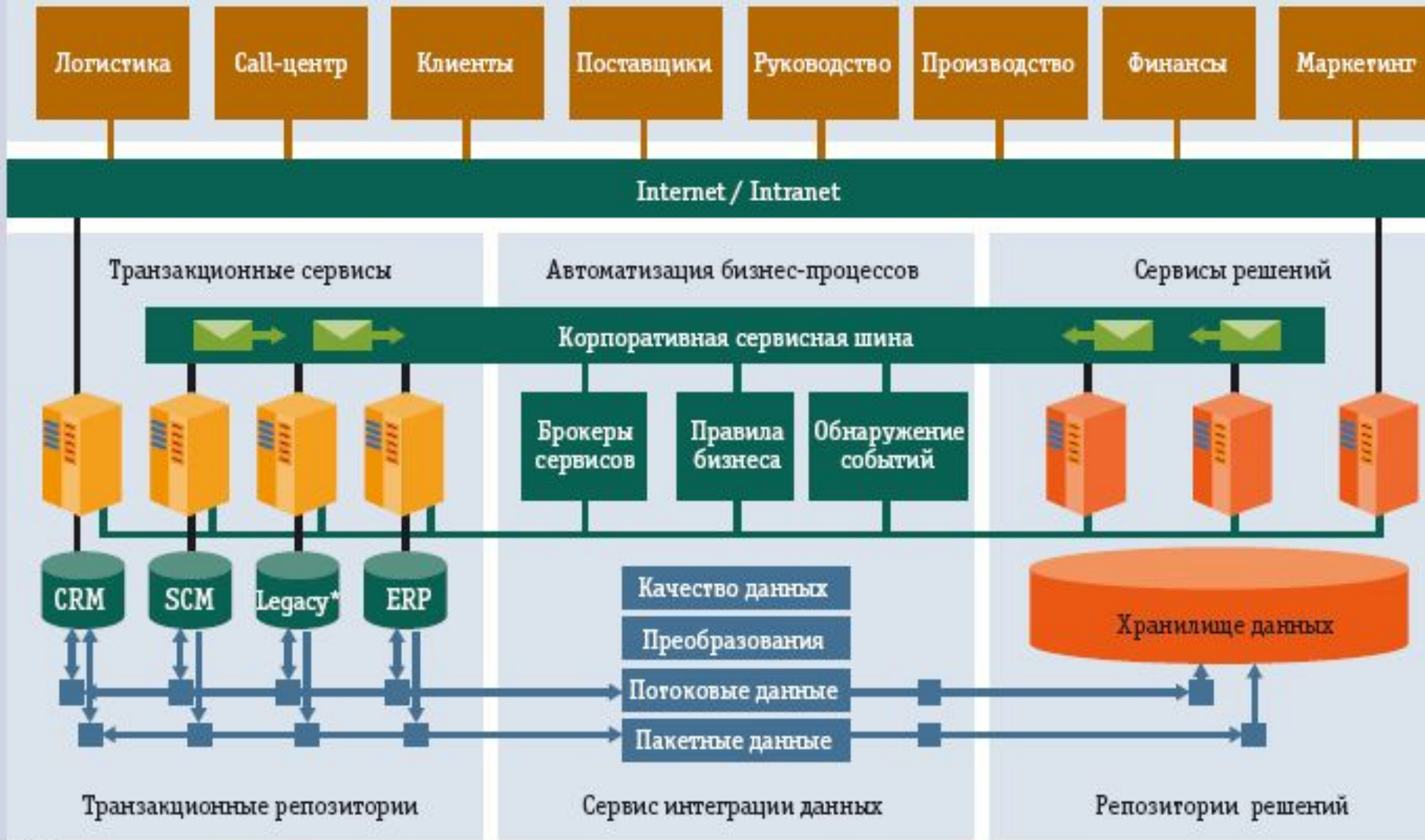
Существуют и другие стандарты описаний, например, спецификация WSCI (Web Services Choreography Interface), которая определяет организацию взаимодействия исполняемых «оркестрированных» процессов, т.е. хореографические аспекты. WSCI описывает наблюдаемое поведение взаимодействующих сервисов: правила упорядочения сообщений, их корреляцию, транзакционные действия, динамику обмена сообщениями и др.

# Интеграционные платформы для реализации идей SOA/BPM/ESB

- **WebSphere Application Server (IBM);**
- **Oracle E-Business Suite (Oracle Corp.);**
- **SAP NetWeaver (SAP AG);**
- **BizTalk Server (Microsoft);**
- **PIE (CMA Small Systems AB);**
- **IFS Applications (IFS);**
- **Universal Applications Service (Siebel);**
- **ИБК «Юпитер».**



# Перспективная архитектура КИС на основе SOA



\* Унаследованные приложения

# Четыре этапа ЖЦ SOA (IBM WebSphere)

(Концепция IBM SOA Foundation)

- **Создание бизнес-модели, ее отладка и моделирование бизнес-процессов (WebSphere Business Modeler);**
- **Сборка системы – разработка КИС и её реализация в рамках SOA-парадигмы (WebSphere Integration Developer);**
- **Внедрение системы - развертывание КИС (WebSphere Process Server);**
- **Управление (WebSphere Business Monitor).**

IBM SOA Foundation – набор идей, инструментариев и конструкций для SOA



# Состав инструментальных средств IBM WebSphere AS

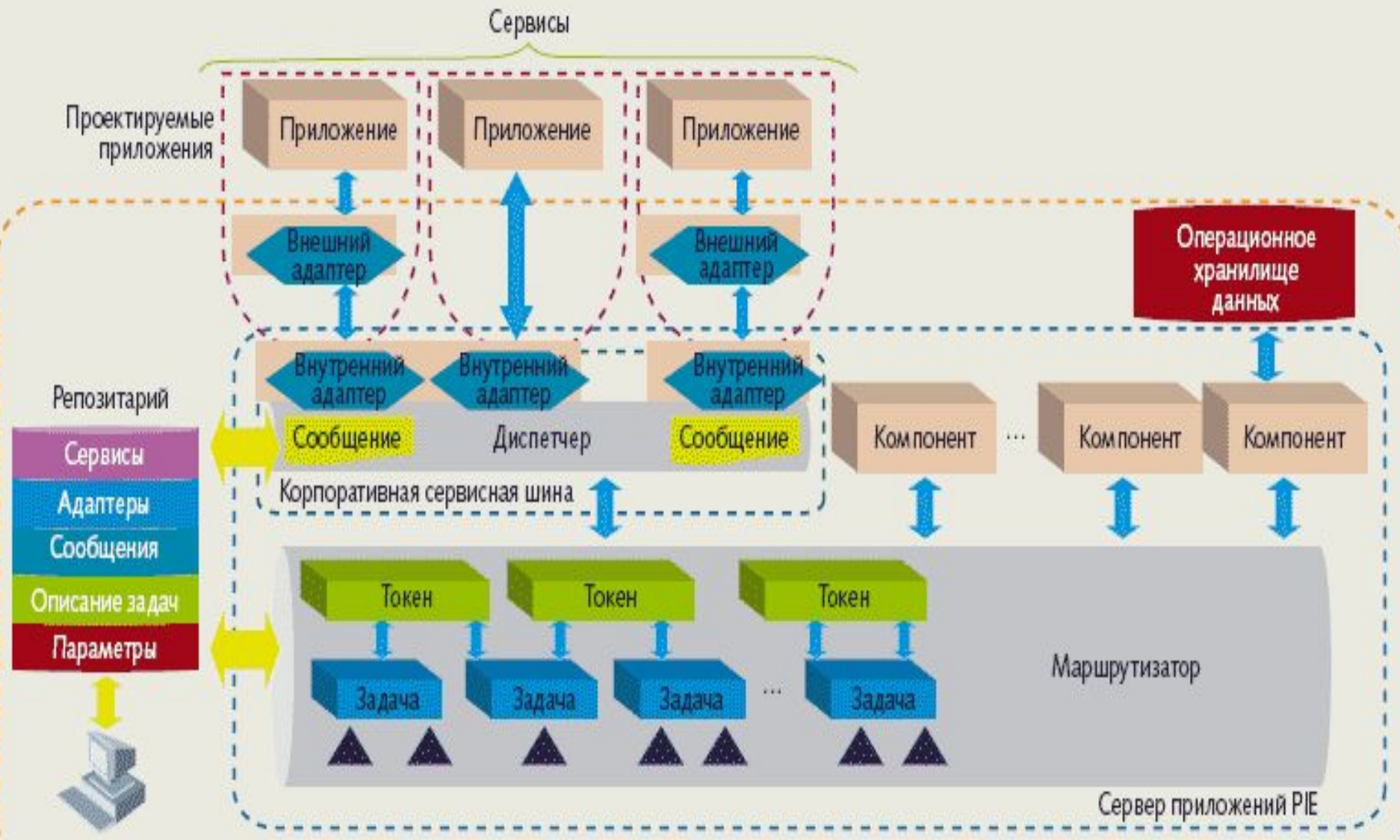
- **WebSphere Business Modeler** – инструментальное средство анализа и моделирования бизнес-процессов;
- **WebSphere Integration Developer** – инструментарий для разработки сервисов в рамках парадигмы SOA и сборки их в исполняемые бизнес-процессы, которые могут быть развернуты на сервере WebSphere P-Server;
- **WebSphere Process Server** – исполнительная среда для развертывания и исполнения сервисных приложений, созданных в рамках WebSphere I-Developer;
- **WebSphere Business Monitor** – средство визуализации в реальном масштабе времени хода выполнения бизнес-процессов с возможностью административного вмешательства и оперативного внесения усовершенствований в реализуемые процессы.

# Основные элементы инструментария WebSphere Integration Developer (IBM)

- Бизнес-объекты (соответствуют структурам данных);
- Интерфейсы (описывают средства и правила взаимодействия компонентов в бизнес-модели и SOA);
- Бизнес-процессы (генерация интегрированных приложений для поддержки бизнес-процессов в виде спецификаций на языке BPEL);
- Пользовательские функции – поддерживаются возможности вмешательства пользователей в логику выполнения бизнес-процесса (при необходимости);
- Машина состояний – поддерживаются механизмы управляемых событиями транзакций для перевода системы из одного состояния в другое (целевое);
- Процедуры – повторяющиеся наборы действий, неоднократно используемые компонентами;
- Бизнес-правила – задаются изначально как условия существования бизнес-компонентов во внешней бизнес-среде;
- Сборщик (assembly editor) – механизм сборки приложений.

# Архитектура системы PIE

(СМА Small Systems AB)

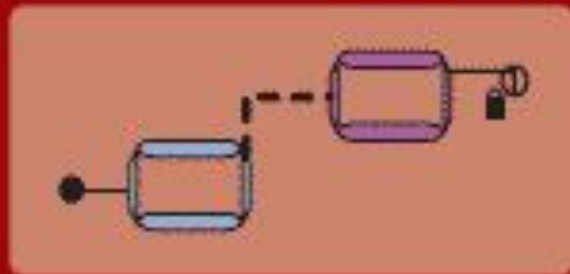


**Processware Integration Environment (PIE)**

PIE

# Уровни представления бизнес-процессов в РІЕ

Метафреймы



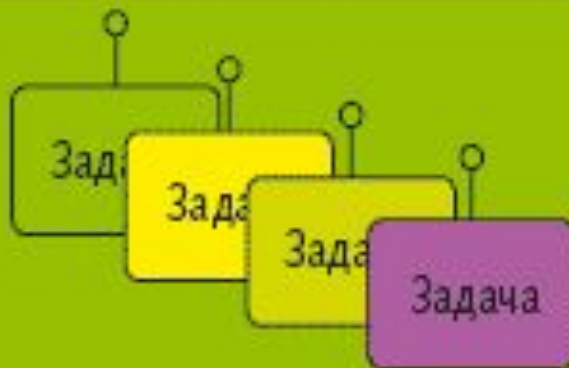
Уровень  
представления  
взаимодействия

Фреймы



Уровень  
бизнес-  
логики

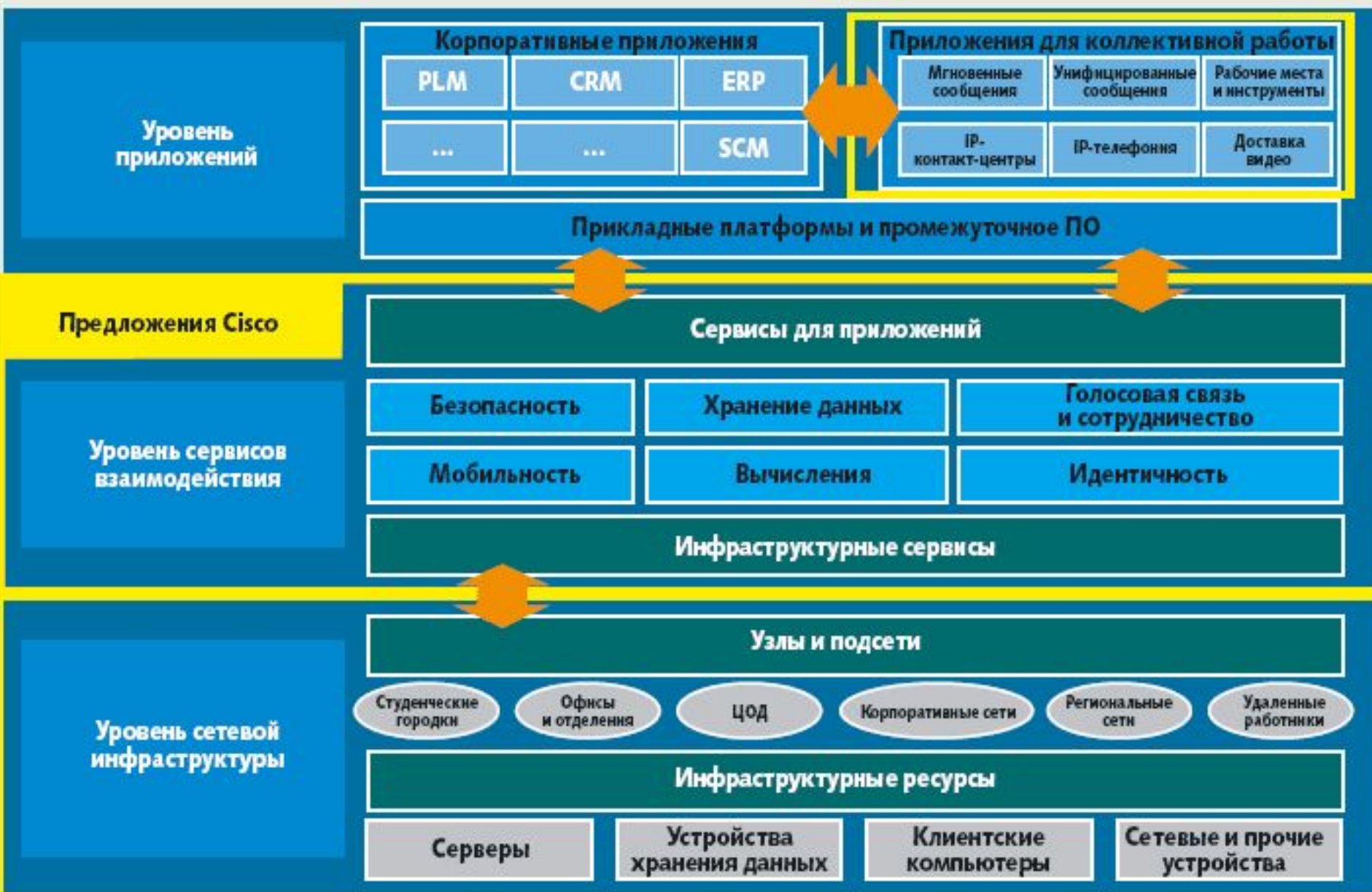
Задачи



Уровень  
программных  
кодов

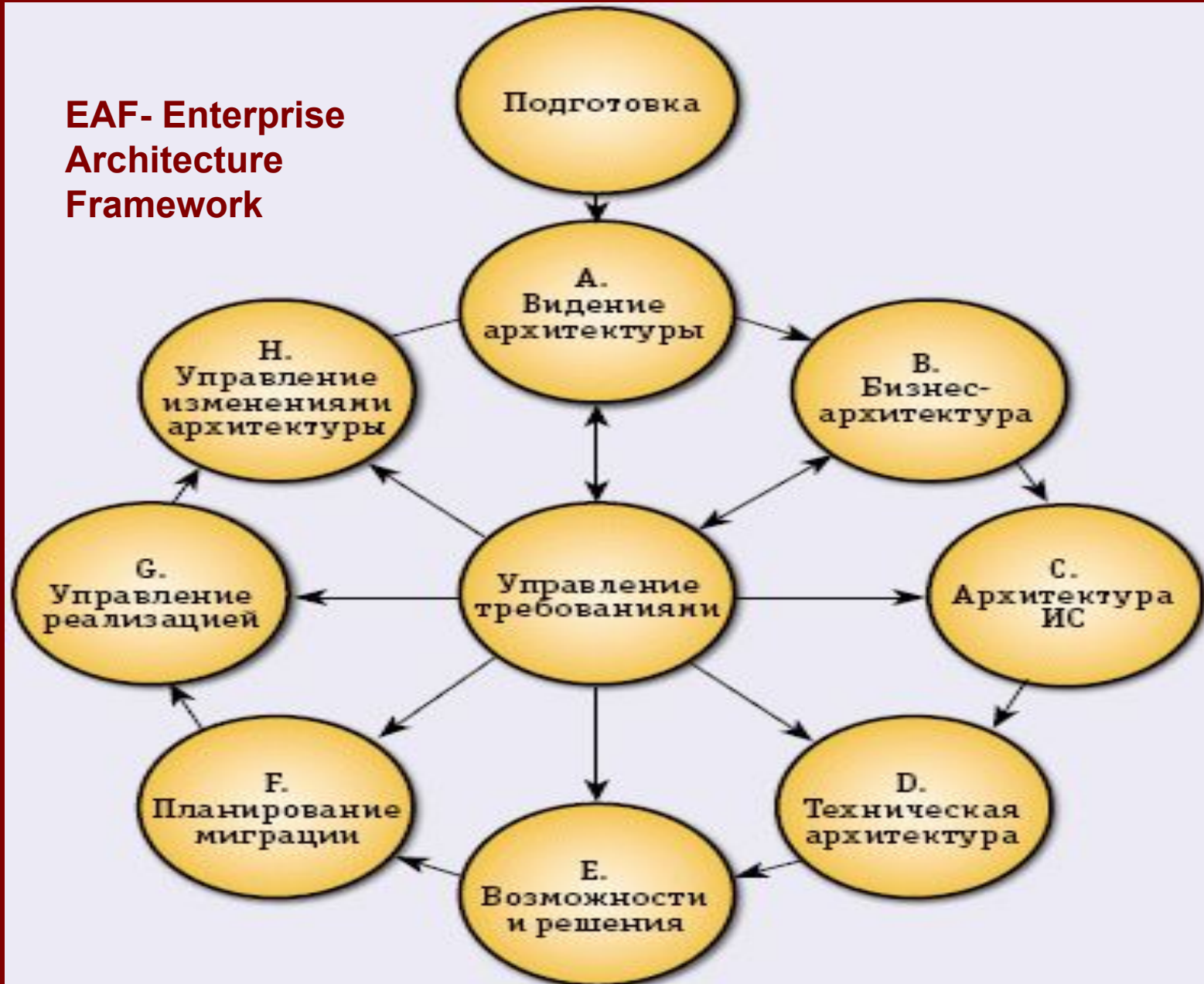


# Сетевая сервисная архитектура SONA Cisco



# Концепция реализации SOA/EA (SAP EAF)

**EAF- Enterprise  
Architecture  
Framework**



# Методология реализации SOA/EA (SAP EAF)

Методология состоит из SAP EAF Architecture Process, SAP EAF MetaModel, руководства, ресурсной базы и инструментов.

SAP EAF Architecture Process – модель, описывающая процесс разработки архитектуры и предлагающая разбить его на фазы: видение архитектуры (A) – определение границ проекта, разработка общего представления архитектуры, согласование плана работ и подхода; бизнес-архитектура (B) – множество бизнес- и технических требований, целей и стратегических направлений развития, данные гар-анализа, описание текущих и целевых бизнес-процессов, функций, сервисов, план перехода на новую бизнес-архитектуру; архитектура информационной системы (C) – формализованное описание существующей архитектуры приложений и данных на основе разработанной бизнес-архитектуры и проведенного гар-анализа, документация по целевой архитектуре и составные элементы архитектуры приложений и архитектуры данных. Далее в SAP EAF Architecture Process входит описание текущей и целевой технической архитектуры (D): системное программное обеспечение, серверы, базы данных, аппаратное обеспечение, сеть, а также высокоуровневый план перехода к целевой архитектуре (E).

# Методология SAP EAF (окончание)

Планирование миграции (F) предполагает разработку детального плана перехода к целевой архитектуре, включающего все необходимые для этого изменения (организационные, бизнес-процессы, ИТ-ландшафт). Основная цель управления реализацией (G) состоит в мониторинге процессов проектирования, реализации корпоративных информационных систем и своевременном внесении изменений во все домены архитектуры предприятия. Управление изменениями архитектуры (H) включает в себя весь комплекс мероприятий по управлению изменениями (разработка новой функциональности, изменения в бизнес-процессах, изменения принципов управления). Управление требованиями является ядром процесса создания архитектуры предприятия на базе SAP EAF и предусматривает постоянный мониторинг, сбор и формализацию потребностей бизнеса.

Модель архитектуры предприятия на базе SAP EAF может быть описана с помощью целого ряда инструментов, существующих сегодня на рынке моделирования бизнес-процессов (например, ARIS). Таким образом, методология SAP EAF определяет последовательность действий (SAP EAF Architecture Process) и набор результирующих документов (матрицы, представления – SAP EAF Metamodel), разрабатываемых на каждом шаге при проектировании архитектуры предприятия.



# ХaaS — что угодно как сервис

**AaaS (Architecture as a Service) — архитектура как сервис**

**BaaS (Business as a Service) — бизнес как сервис**

**DaaS (Data/Documents as a Service) — данные/документы как сервис**

**DISaaS (Data Integration System as a Service) — интеграция данных как сервис**

**EaaS (Ethernet as a Service) — Ethernet как сервис**

**FaaS (Firmware as a Service) — конструкция для разработки и внедрения приложений, поставляемая как сервис**

**GaaS (Globalization as a Service) — глобализация как сервис**

**Haas (Hardware as a Service) — аппаратное обеспечение как сервис**

**IaaS (Infrastructure/Information as a Service) — инфраструктура/ информация как сервис**

**IDaaS (Identification as a Service) — идентификация как сервис**

**PaaS (Platform as a Service) — платформа как сервис**

**OaaS (Organization/Optimization/Operations as a Service) — организация/ оптимизация/операции как сервис**

**TaaS (Technology as a Service) — технологии как сервис**

**VaaS (Voice as a Service) — голос как сервис.**

# Решение проблем информационной безопасности в рамках SOA

# Стандарты безопасности в области SOA

**WS-Security** поддерживает безопасность протокола SOAP, конфиденциальность и целостность сервисов; согласован с другими средствами авторизации подтверждения, такими как сертификаты X.509, Kerberos и SAML.

**Security Assertion Markup Language (SAML)** служит для обмена подтверждениями об аутентификации и авторизации между доменами (см. [www.opensaml.org](http://www.opensaml.org)). В русском языке не установился общепринятый перевод термина SAML; поэтому его трактуют и как «язык разметки утверждений безопасности», и как «язык разметки допуска к информации», и как «язык разметки для инфраструктуры безопасности». Каждый из переводов по-своему корректен. Спецификация Liberty Alliance 1.0 уточняет SAML.

**XML Access Control Markup Language (XACML)**, язык разметки управления доступом, является расширением XML для создания конструкций, служащих для автоматизации авторизации и управления доступом к информации. Средства XACML определяют те директивы, которые формируются и передаются для управления доступом.

# Стандарты безопасности в области SOA (окончание)

**Extensible Rules Markup Language (XrML)**, язык управления правами собственности в мультимедиа, порожден движением Digital Rights Management и в какой-то мере составляет конкуренцию языку XACML. **XML Encryption** и **XML Digital Signature**, стандарты криптографии и цифровой подписи.

**Service Provisioning Markup Language (SPML)** задуман как средство описания вспомогательных данных, необходимых для установления отношений между сервисами. Разрабатывается ассоциацией OASIS для обмена данными между кооперирующимися организациями (см. также [www.openspml.org](http://www.openspml.org)).

**XML Key Management Specification (XKMS)** служит стандартом для инфраструктуры открытых ключей **XKISS (XML Key Information Service Specification)** и регистрационной спецификации **XKRSS (XML Key Registration Service Specification)**. Спецификация призвана упростить обмен ключами по схеме, предложенной Диффи и Хеллманом.

# Основные механизмы обеспечения информационной безопасности

- Экранирование (физическое и программное);
- Управление доступом (авторизация, приоритезация и др.);
- Идентификация и аутентификация;
- Протоколирование и аудит (регистрация и учет);
- Криптография (шифрование).

# Платформа для цифровой идентификации (identity framework)

- *integrity and non-repudiation* — целостность и невозможность отказа от авторства посланного сообщения;
- *confidentiality* — конфиденциальность;
- *authentication and authorization* — аутентификация и авторизация;
- *identity provisioning* — обеспечение идентичности;
- *representing and managing authorization policy* — представление и управление политикой авторизации.

# Контекст проблемы идентификации для SOA

Управление доступностью контента

Отношения между людьми, коллективами и организациями

Определение отношений на основе опыта

Определение отношений между субъектами

Аутентификация цифровой  
идентичности сетевых субъектов

Персонализация доступных субъектам областей

Управление доступом и аутентификацией

Бизнес-политики — ответственность и страховка транзакций

Законодательство и публичная политика

# «Цифровая» идентичность субъекта

Идентичность субъекта представляется в виде набора данных, которые содержат его атрибуты, предпочтения и особые черты. Атрибуты могут быть стандартными; в приложении к человеку это записи в медицинской карте, истории покупок или кредитные истории, размеры одежды, рост и т. д. Особые черты —особенности корпоративной истории предприятия. Подобная систематика атрибутов применима и к объектам любой природы. На техническом уровне цифровую идентичность можно представить как набор признаков сетевого субъекта, зафиксированный в виде электронных записей.

Для того чтобы сетевой субъект мог воспользоваться нужным ресурсом, он должен предъявить мандат (удостоверение), который подтверждает право на такое использование. Мандат позволяет установить доверительные отношения между поставщиком и потребителем сервисов. Его предъявляют на контрольном пункте службе безопасности для аутентификации этого мандата, глубина которой должна соответствовать риску доступа к ресурсу. Затем служба безопасности назначает для сетевого субъекта политику защиты, в соответствии с которой следующая инстанция, пункт реализации политики, определяет наименования доступных ресурсов и глубину доступа к ним. Такие сведения возвращаются в службу безопасности, обеспечивающую защиту данных от несанкционированного доступа, разрушения или изменения.



# Что такое IDaaS

Идея IDaaS возникла из стремления абстрагировать функции управления идентификацией от конкретных бизнес-приложений и решений по информационной безопасности, предоставляя их в доказавшей свою эффективность архитектуре слабосвязанных прикладных сервисов. Такие сервисы реализуются независимо от бизнес-приложений и друг от друга, могут неоднократно использоваться и предоставляться приложениям или пользователям как в рамках инфраструктуры одной компании, так и, при наличии соответствующей интеграции (B2B), за ее пределами.

IDaaS определяют как *набор многократно используемых, стандартизированных сервисов, которые предоставляют приложениям различные функции управления идентификацией и доступом.*

По существу, IDaaS представляет собой систему функциональных компонентов, которые реализуются в четырех-пяти традиционных, перекрывающихся по своим возможностям приложениях управления идентификацией и доступом, в том числе решениях по обеспечению однократной регистрации (Single Sign-On, SSO), системах доставки (provisioning) идентификационных данных и сервисах каталогов. Благодаря IDaaS эти компоненты реализуются независимо друг от друга и предоставляются приложениям или конечным пользователям (посредством порталов) по запросу для осуществления функций управления идентификацией и правами доступа.

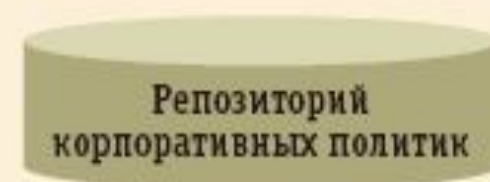
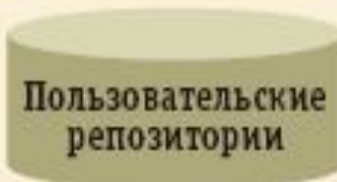
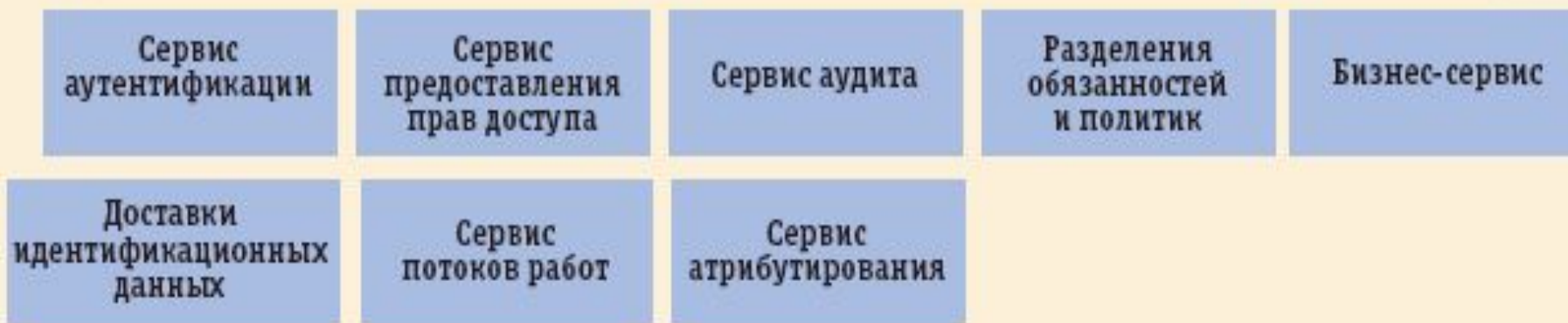
Сервисы в IDaaS могут использоваться отдельно или включаться в согласованные процессы управления идентификацией и контроля доступа с помощью средств оркестровки на базе корпоративной сервисной шины (ESB).

# Идентификация как сервис (IdaaS)

## Инфраструктура IdaaS



## Корпоративная сервисная шина



# Виды сервисов в модели IDaaS

**Сервисы аутентификации и общей авторизации (*authentication and coarse-grained authorization services*). Аутентификация пользователей в корпоративных приложениях — основное требование в любой защищенной среде. К сервисам аутентификации и общей авторизации будут обращаться практически все системы в компании с целью получить ответ на вопрос — может ли пользователь с данным набором удостоверений личности (имя пользователя и пароль, а также, возможно, еще какая-либо дополнительная информация в системах с многофакторной аутентификацией) получить доступ к запрашиваемому им ресурсу.**

**Сервисы предоставления прав доступа (*entitlement services*). Сервисы этой категории обеспечивают приложения функциями более детальной авторизации. С помощью сервисов предоставления прав доступа приложения получают информацию о том, к какой конкретно их функциональности пользователь может обращаться. Таким образом, сотрудник, имеющий общий доступ, например, к системе SAP, может получить дополнительные привилегии по использованию ресурсов системы в зависимости от своей роли в компании. Потребителями сервисов данного типа будут решения по управлению корпоративными политиками в совокупности с приложениями, поддерживающими деятельность определенных бизнес-подразделений.**

# Виды сервисов IDaaS (продолжение)

*Сервисы доставки идентификационных данных (provisioning services).*

Система доставки идентификационных данных отвечает за предоставление сотруднику определенных прав доступа к корпоративным ресурсам, когда он поступает на работу, и их изменение, когда меняется его статус.

Соответствующие сервисы в IDaaS обеспечат управление полным жизненным циклом идентификационных данных сотрудников, партнеров и клиентов компании в их пользовательских репозиториях и будут востребованы как внутренними корпоративными приложениями, так и внешними решениями.

*Сервисы потоков работ (workflow services).* Запросы сотрудников на доступ к тем или иным информационным ресурсам утверждаются менеджерами компаний. Сервисы потоков работ автоматизируют процессы утверждения и другие последовательности операций по получению доступа к приложениям и предоставляют возможность различным системам обращаться к процессам управления идентификационными данными. Когда пользователь запрашивает новые права доступа к приложению, приложение вызывает сервис потока работ и передает ему запрос. Система управления потоками работ генерирует сообщение для менеджера о появлении нового запроса, требующего утверждения. Когда менеджер входит в систему, приложение вызывает сервис потоков работ, для того чтобы оповестить менеджера о запросах, которые необходимо утвердить. После того как менеджер утвердит запрос, пользователь сможет получить доступ к приложению.

# Виды сервисов IDaaS (продолжение 2)

**Сервисы атрибутирования (*attribute services*).** Часто приложениям нужна информация о пользователях, размещенная в разных источниках, и в этом случае они обращаются к сервисам атрибутирования, которые выполняют роль информационных сервисов IaaS в приложении к идентификационным данным пользователей, агрегируя атрибуты идентификации из множества различных источников. Механизмом реализации сервисов атрибутирования могут быть так называемые «сервисы виртуальных каталогов» (*virtual directory*), которые консолидируют информацию о пользователях из традиционных LDAP-каталогов и баз данных, представляя их приложениям как единый источник идентификационных данных (пример — система Oracle Virtual Directory).

**Сервисы аудита (*auditing services*).** Необходимость выполнять положения определенных нормативных актов требует проведения тщательного аудита доступа к корпоративным приложениям и агрегирования соответствующей информации. Сервисы аудита реализуют единый механизм сбора и предоставления информации о регистрации пользователей в корпоративных системах для формирования общей отчетности.

# Виды сервисов IDaaS (окончание)

*Сервисы разделения обязанностей и политик (segregation of duties and policy services).* Политики, определяющие принятие тех или иных решений относительно идентификации и прав доступа, могут управляться из различных источников. Данная категория сервисов предоставляет единый интерфейс всем приложениям, которые запрашивают политики и управляют ими, определяя, не вступает ли осуществляемый пользователем доступ в конфликт с принятыми правилами разделения обязанностей (механизмы разделения обязанностей гарантируют предоставление определенных, непротиворечивых привилегий конкретным категориям пользователей). Сервисы политик также могут быть использованы приложениями для получения функций управления ролями в компании.

*Бизнес-сервисы.* Сервисы управления идентификацией могут использоваться в модели предоставления программного обеспечения как услуги (SaaS) или в среде информационных сервисов (IaaS) для реализации функций защиты данных

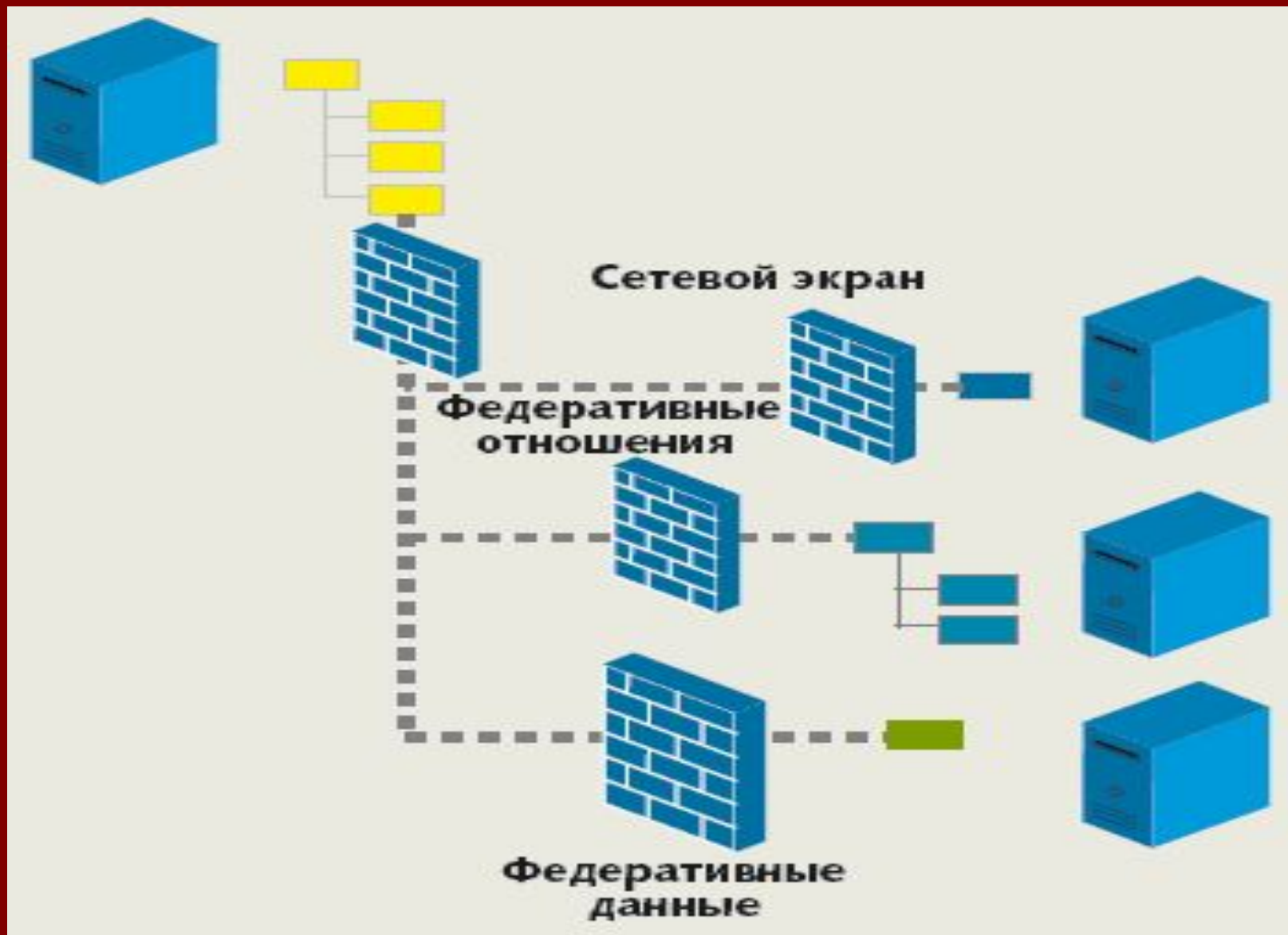
# Федеративное управление IDaaS (Federated Identity Management - FIM)

Федерация — это подход, позволяющий Web-приложениям в условиях существования слабосвязанных архитектур SOA передавать идентификационные подтверждения из одного домена в другой, а благодаря этому — избегать избыточных действий, вызываемых повторной локальной аутентификацией.

Федеративный подход поддерживается двумя основными идентификационными стандартами: службой WS-Security и языком SAML.



# Архитектура федеративных отношений





# Краткая характеристика FIM

FIM позволяет организациям взаимодействовать между собой в защищенном режиме. Данное решение реализует слабосвязанную модель управления идентификацией пользователей и доступом к ресурсам, размещенным в нескольких компаниях или защищенных зонах. Например, в случае двух взаимодействующих компаний решение FIM не реплицирует структуры управления идентификацией и безопасностью обеих организаций, а предоставляет общую модель для единого управления идентификацией и обеспечивает доступ к информации и сервисам на основе доверительных отношений между ними. Компаниям, применяющим SOA и Web-сервисы, решение FIM обеспечивает основанное на правилах единое управление безопасностью для федеративных Web-сервисов.

Основа FIM — доверительные отношения, целостность и конфиденциальность данных. Это дает организациям возможность совместно использовать идентификационные данные и правила доступа пользователей к сервисам, не дублируя локальные идентификационные данные и правила безопасности. Совместное использование идентификаторов и правил безопасности в рамках «федерации» (объединения партнеров на условиях взаимного доверия) — ключевое условие для предоставления сотрудникам расширенных возможностей перемещения между несколькими объединенными сайтами этой федерации. Доверительные отношения позволяют компаниям реализовать нежесткое объединение применяемых в каждой компании систем управления идентификацией пользователей.

# Преимущества модели FIM

- Упрощение интеграции между сайтами компании и ее бизнес-партнеров, включая управление сессиями;
- Улучшение соответствия бизнес-требованиям за счет ослабления угроз безопасности;
- Расширение возможностей пользователей благодаря технологии централизованного входа в систему (SSO);
- Упрощение администрирования безопасности в межкорпоративных бизнес-процессах на основе «сервисов безопасности»;
- Интегрированное управление безопасностью на основе правил для Web-сервисов в SOA-среде;
- Поддержка открытых стандартов и спецификаций, включая LAP, SAML, WS-Federation, WS-Security и WS-Trust.

# Стороны-участники

## модели федеративных удостоверений

Всякий раз, когда производится проверка удостоверений, федеративная модель предусматривает участие четырех логических компонентов:

- *Пользователь* — лицо, которому назначается определенное удостоверение для взаимодействия с оперативным сетевым приложением.
- *Агент пользователя* — браузер или другое приложение, выполняемое на любом оборудовании, от ПК до мобильного телефона или медицинского устройства. Оперативные взаимодействия пользователя всегда проходят через агента, который или пассивно пропускает поток идентификационной информации, или активно управляет им.
- *Сайт поставщика услуг (SP)* — Web-приложение (например, приложение подготовки отчетов о затратах), делегирующее проверку подлинности третьей стороне, которая также может переслать поставщику услуг некоторые атрибуты пользователя. Поскольку поставщик услуг использует внешнюю информацию, его часто называют «проверяющей стороной» (relying party, RP).
- *Поставщик удостоверений* (identity provider, IdP, иногда используется термин «провайдер идентификации») — Web-сайт, на котором регистрируются пользователи и чаще всего хранятся их атрибуты, представляющие общий интерес для разных поставщиков услуг.

# Стандартизация в рамках FIM

(Identity and Access Management, IAM )

**WS-Security** поддерживает безопасность протокола SOAP, конфиденциальность и целостность сервисов; согласован с другими средствами авторизации подтверждения, такими как сертификаты X.509, Kerberos и SAML.

**Security Assertion Markup Language (SAML)** служит для обмена подтверждениями об аутентификации и авторизации между доменами.

**XML Access Control Markup Language (XACML)**, язык разметки управления доступом, является расширением XML для создания конструкций, служащих для автоматизации авторизации и управления доступом к информации. Средства XACML определяют те директивы, которые формируются и передаются для управления доступом.

**Extensible Rules Markup Language (XrML)**, язык управления правами собственности в мультимедиа, является конкурентом языка XACML.

**XML Encryption** и **XML Digital Signature** - стандарты криптографии и цифровой подписи.

# Стандартизация Identity Management (окончание)

**Service Provisioning Markup Language (SPML)** задуман как средство описания вспомогательных данных, необходимых для установления отношений между сервисами. Разрабатывается ассоциацией OASIS для обмена данными между кооперирующимися организациями.

**XML Key Management Specification (XKMS)** служит стандартом для инфраструктуры открытых ключей **XKISS (XML Key Information Service Specification)** и регистрационной спецификации **XKRSS (XML Key Registration Service Specification)**. Спецификация призвана упростить обмен ключами по схеме Диффи-Хеллмана.

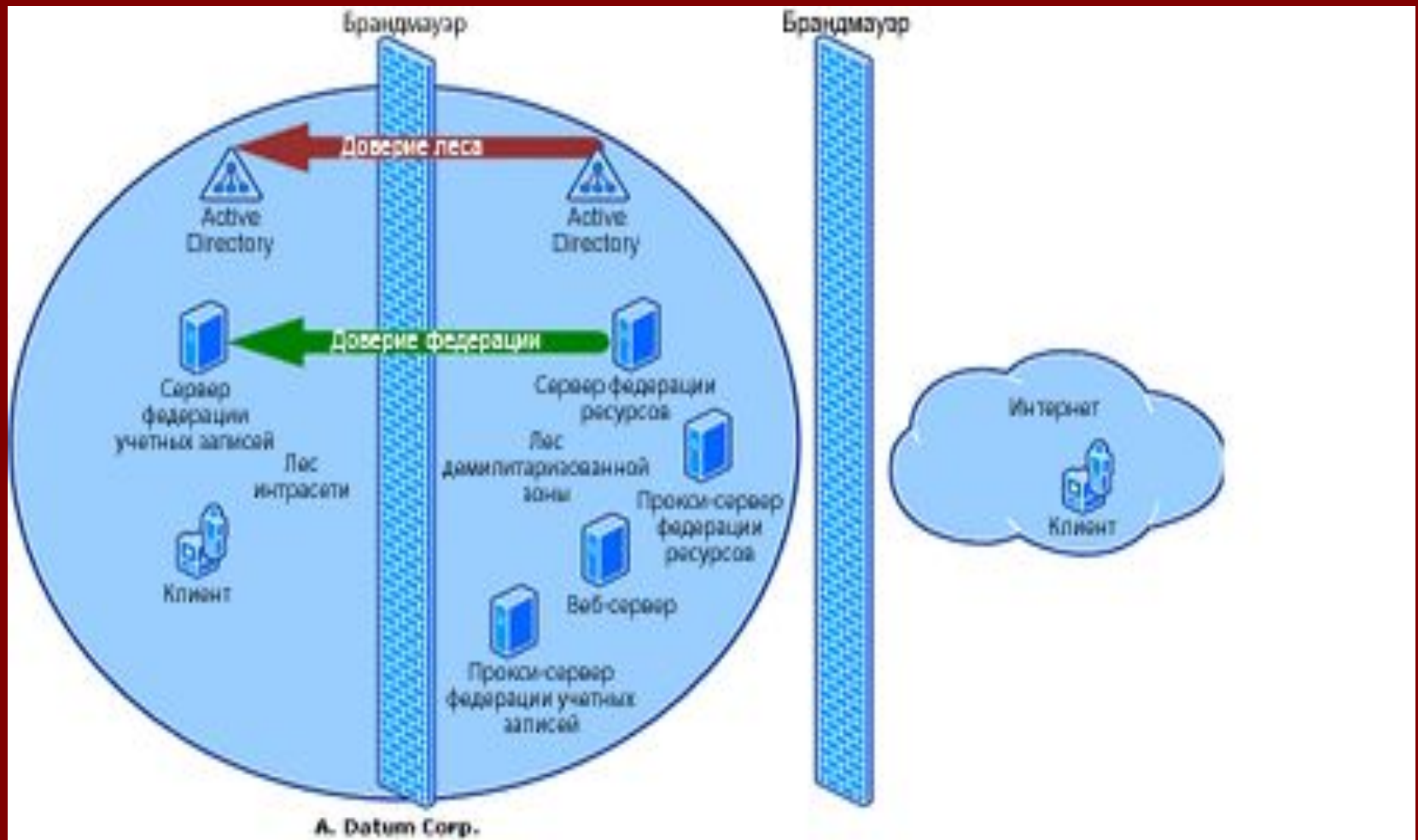
За исключением языка SAML, перечисленные стандарты пока не получили однозначного признания. Например, Microsoft ориентируется на использование Kerberos-билетов (Kerberos ticket) и подтверждений XrML (XrML assertion) в подтверждающих токенах (assertion token) для Passport и TrustBridge, а другие компании ориентруются на SAML. Достигнута договоренность между IBM, Microsoft и Sun Microsystems о совместной поддержке комбинации SOAP, WS-Security и SAML.

# Федеративная система удостоверений (централизация функций управления ИБ)

*Федеративное управление удостоверениями* — набор технологий и процессов, посредством которых компьютерные системы динамически распространяют сведения об удостоверениях и делегируют соответствующие задачи по доменам безопасности.

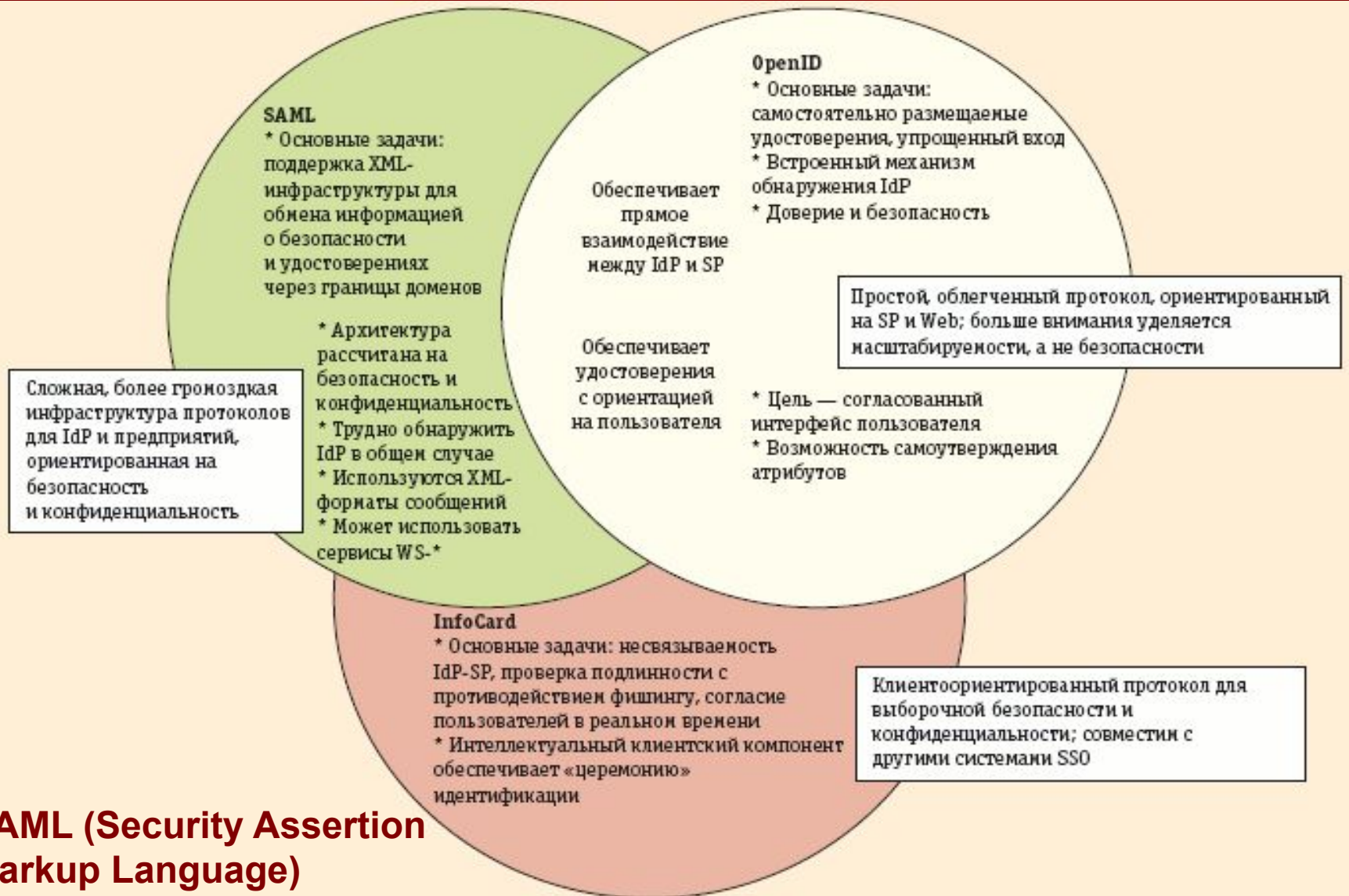
Федеративное удостоверение — средство, с помощью которого Web-приложения обеспечивают однократный вход между доменами (Single-Sign On, SSO); в результате пользователи могут пройти проверку подлинности один раз и получить доступ к защищенным ресурсам и сайтам в других доменах.

# Реализация федеративной службы FIM





# Протокольные службы идентификации

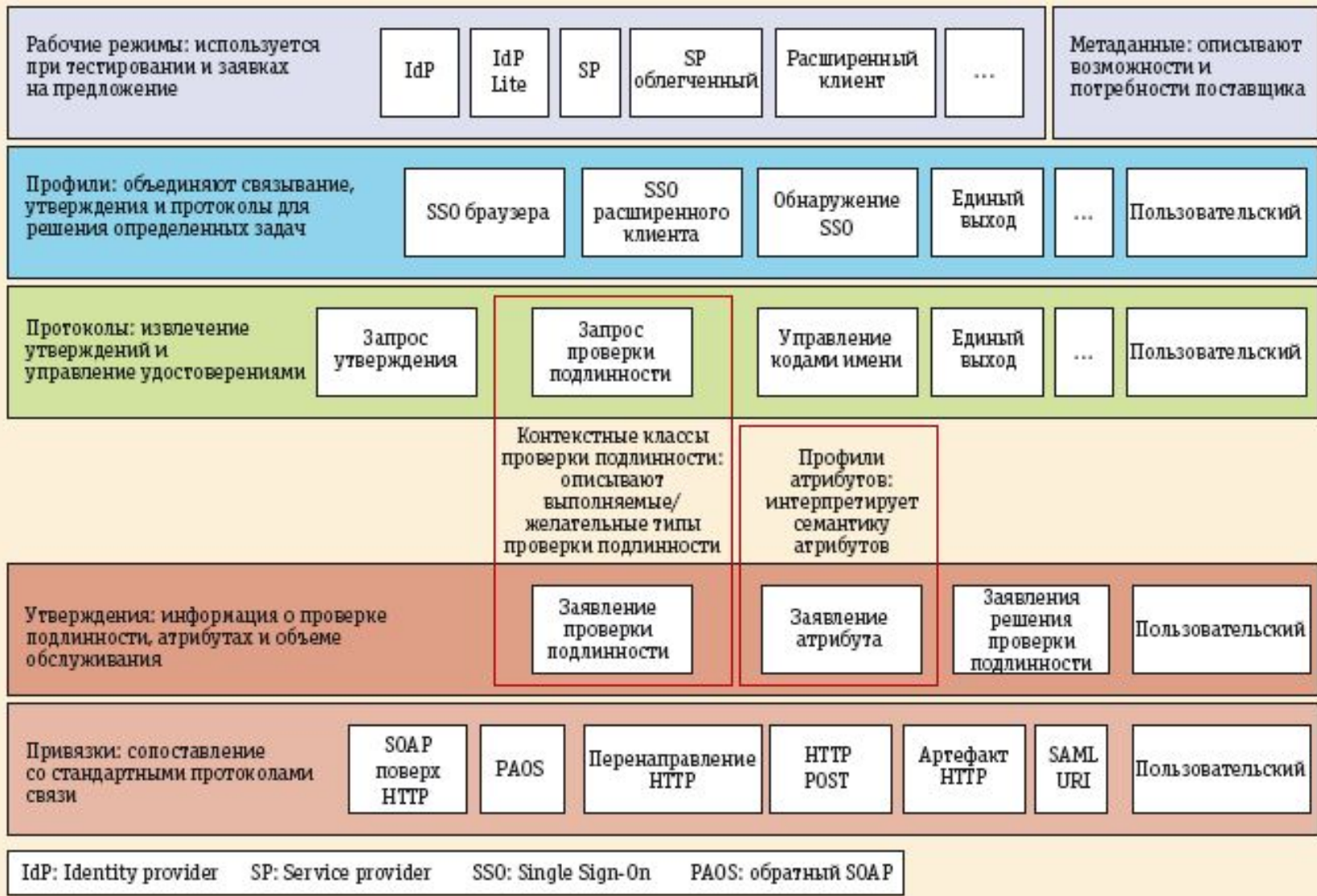


## SAML (Security Assertion Markup Language)

Три распространенных протокола федеративных удостоверений SAML, OpenID и InfoCard имеют как общие черты, так и явные различия



# Инфраструктура стандарта SAML



Инфраструктура SAML. Утверждения SAML состоят из XML-пакетов, которые содержат такую информацию, как идентификатор, состояние проверки подлинности и атрибуты целевого пользователя

# Организация стандарта SAML

**SAML — стандарт OASIS и ITU (ITU-T X.1141), который обеспечивает XML-инфраструктуру для обмена информацией о безопасности и удостоверениях через границы доменов. SAML — нечто вроде универсального раствора удостоверений, а его архитектура удовлетворяет различным нуждам, в том числе поддерживает неодушевленных держателей удостоверений. Его структура определяется строгими требованиями к доверию, транзакциям высокой ценности и конфиденциальности. Несмотря на гибкость инфраструктуры и использование некоторых его компонентов другими технологиями (в том числе CardSpace и различными расширениями OpenID), SAML обеспечивает собственные решения для типовых задач.**

**Ядро SAML состоит из «утверждений» (assertion), то есть XML-пакетов, составленных из идентификатора держателя удостоверения, состояния проверки подлинности и атрибутов.**

**Утверждения и сообщения протокола можно подписывать, шифровать и объединять в профили.**

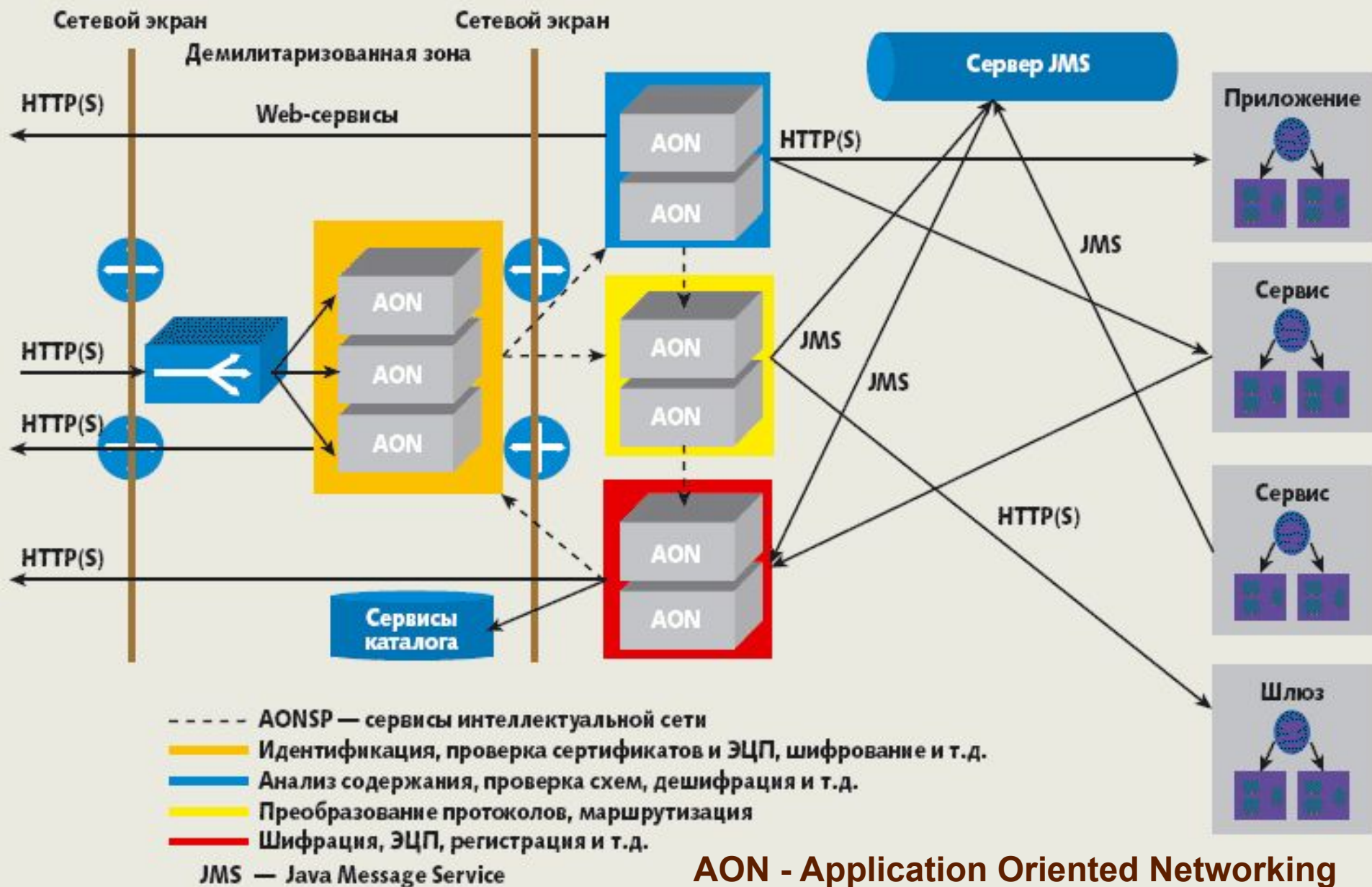
# Возможности стандарта SAML

SAML обеспечивает широкий диапазон решений для процедур единого входа, иницируемых поставщиком услуг и поставщиком удостоверений, связывание учетных записей через федеративный идентификатор, единый выход, обмен атрибутами и долгосрочное управление федеративным идентификатором. Технология снижает вероятность многих угроз для безопасности и конфиденциальности, в частности, предоставляя псевдонимы в нескольких формах. SAML стыкуется со стандартом Identity Web Services Framework (ID-WSF), который охватывает задачи для автономных пользователей и Web-служб на основе соединений. В состав ID-WSF входит настраиваемая служба взаимодействия.

SAML обеспечивает решение для обнаружения поставщика удостоверений на основе cookie общего домена. SAML развертывается в кругах доверия с привязкой к центральному поставщику удостоверений, представляющему крупное сообщество пользователей (например, пользователей КИС), и определенным набором доверенных радиальных поставщиков услуг. В этой ситуации администраторы могут внести информацию IdP в сайты поставщиков услуг до начала взаимодействия SAML, и пользователям будет предоставлен единый вход в КИС.

# Организация стандарта OpenID

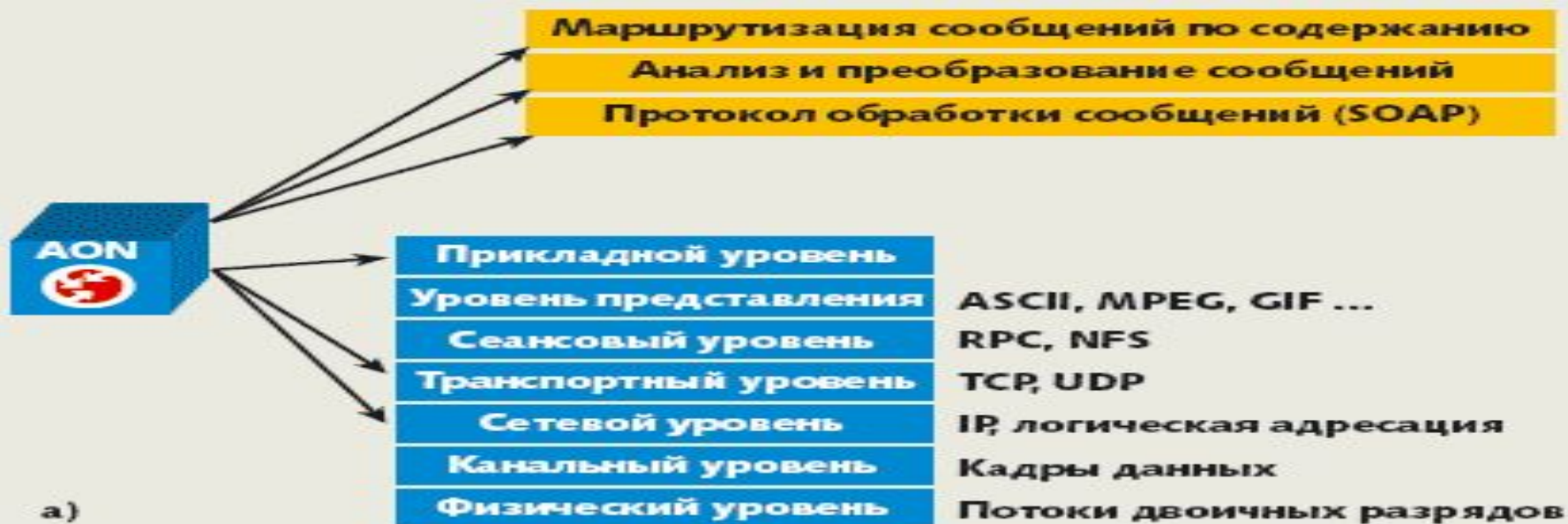
# Архитектура защищенной интеллектуальной SOA-сети (SONA Cisco)



AON - Application Oriented Networking



# Взаимосвязь моделей AON-SONA и ISO/OSI



a)

## Стек протоколов Web-сервисов



б)

# Платформы для реализации FIM (IAM)

- **SOA Security Federation Manager (CA);**
- **IBM Tivoli Federated Identity Manager (IBM);**
- **Oracle Identity Federation (Oracle Corp.);**
- **Oracle Web Services Manager (Oracle Corp.);**
- **SUN Identity Management (SUN Microsystems);**
- **Microsoft Identity Integration Server 2003 (Microsoft);**
- **RSA Federation Identity Manager (RSA-ECM);**
- **ИБК «Юпитер».**

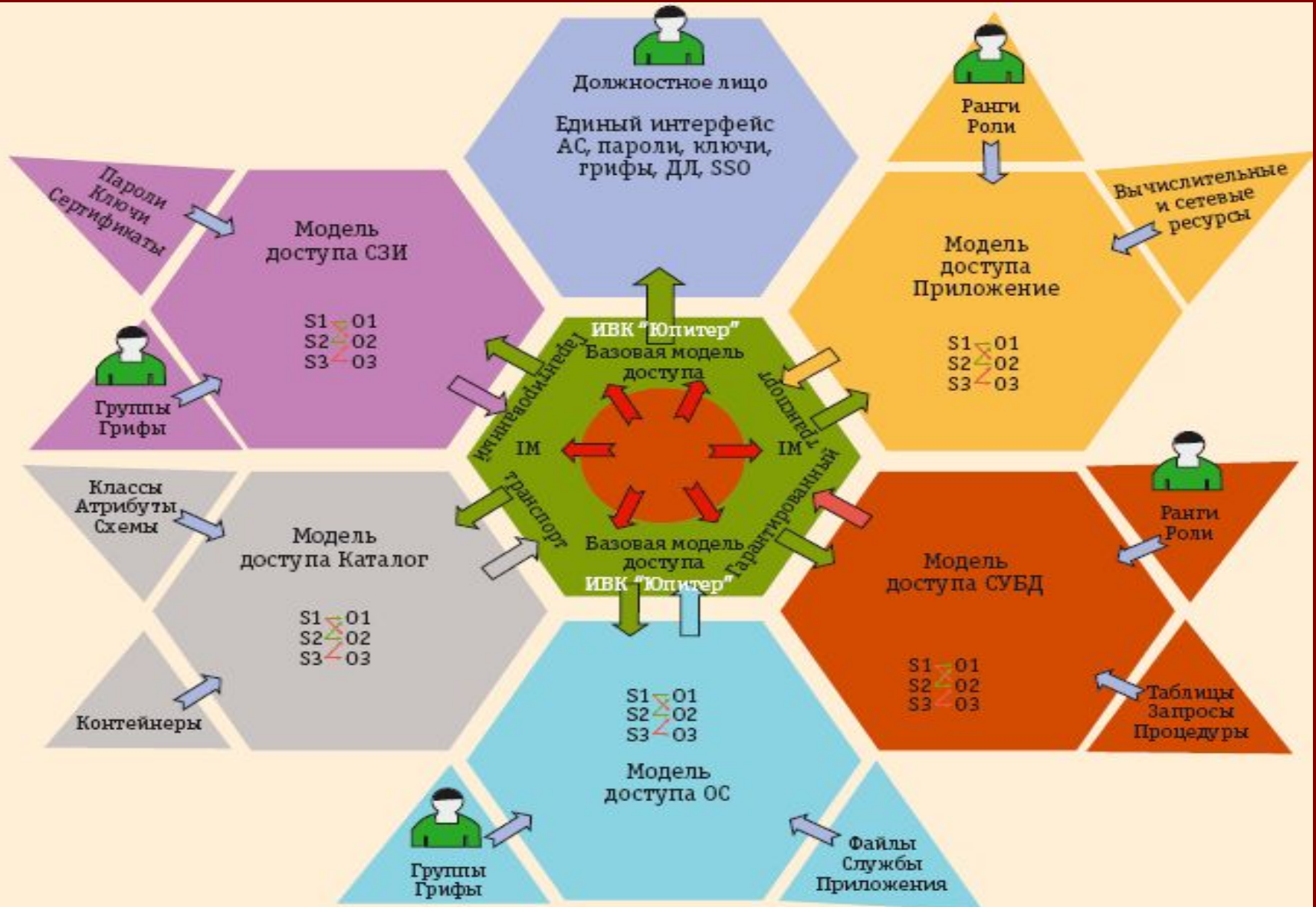


# Oracle Web Services Manager

**Oracle Web Services Manager (OWSM) — комплексное решение для включения средств безопасности и управления на основе политик в существующие или вновь создаваемые Web-сервисы для развертывания решений сервис-ориентированной архитектуры (SOA). Оно позволяет ИТ-менеджерам централизованно задавать политики для управления работой Web-сервисов (политики доступа, аудита и проверки содержимого SOAP-пакета), а затем применять их к Web-сервисам.**

**С помощью этого средства можно также собирать статистику о работе различных компонентов распределенных систем и представлять их в виде наглядных Web-панелей. Ключевые возможности продукта — управление доступом к Web-сервисам и однократная аутентификация, централизованное управление политикой безопасности, унификация процесса мониторинга, а также маршрутизация запросов к Web-сервисам.**

# Интеграционная платформа «ИВК ЮПИТЕР» для SOA



# Функции и задачи платформы «ИВК ЮПИТЕР»

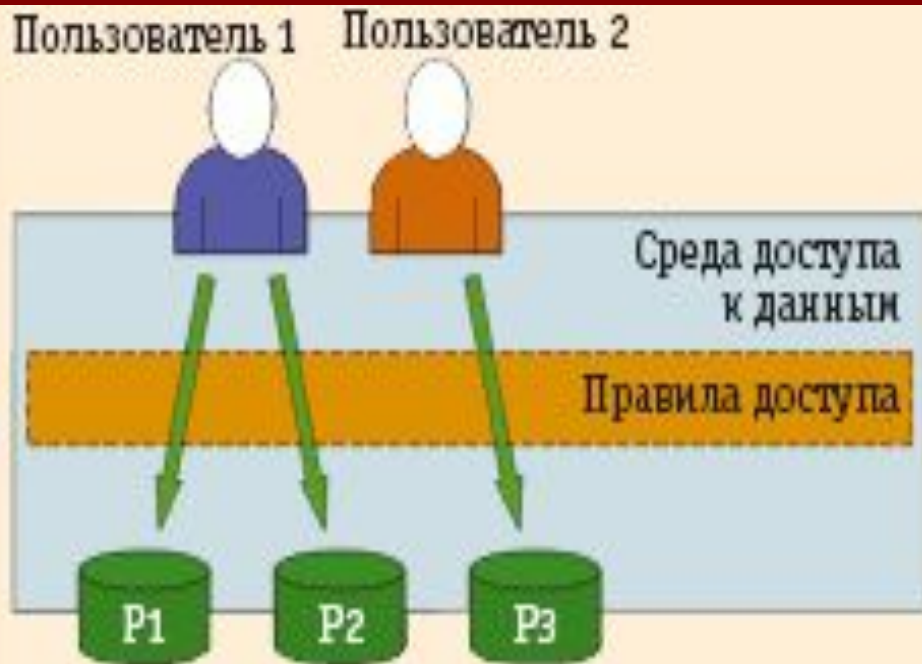
Интеграционная платформа «ИВК Юпитер» представляет собой защищенную реализацию промежуточного ПО, ориентированного на передачу сообщений (Message-oriented Middleware, MOM), являющегося основой для построения SOA. Основные функции MOM «ИВК Юпитер»: гарантированное доведение информации между участниками обмена в информационных системах, наличие синхронного и асинхронного режимов обмена данными, прозрачная система адресации.

Система является сертифицированным средством защиты информации и работает в ИС, обрабатывающих информацию до уровня 1Б включительно (по РД ФСТЭК РФ). Важным шагом на пути решения задачи интеграции систем защиты, встроенных в различные прикладные программные средства и установки корреляции между их контекстами безопасности, явилась разработка программного продукта «ИВК Юпитер. Identity Manager».

# Особенности реализации «ИВК ЮПИТЕР»

«ИВК Юпитер. IM» представляет собой клиент-серверное приложение. Сервер обеспечивает хранение идентификационных параметров и выдачу их по запросу клиентов и совмещается с сервером безопасности «ИВК Юпитер». Клиент «ИВК Юпитер. IM» — оконное приложение, которое запускается автоматически после успешной авторизации должностного лица в магистрали «ИВК Юпитер». Взаимодействие между клиентом и сервером происходит посредством защищенной среды передачи данных «ИВК Юпитер», осуществляющей гарантированное доведение произвольной, в том числе учетной и ключевой, информации и взаимодействия с разнородными *удостоверяющими центрами*, сертификатами которых должны быть подписаны отправления конкретных приложений. Автоматизированное подписание необходимым сертификатом, хранящимся в хранилище «ИВК Юпитер», всего трафика для каждого выделенного приложения — новая особенность функционирования разнородных информационных систем, включающих в себя разнородные приложения, связанные с различными *удостоверяющими центрами*.

# Механизмы передачи контекста безопасности в SOA («ИВК Юпитер»)





# Требования к ESB

## в разрезе информационной безопасности

- Реализация системы управления правилами разграничения доступа, охватывающей все базовые средства доступа к первичным информационным ресурсам (файловая система, СУБД и т.п.) и обеспечивающей доступ из единого центра управления безопасностью к специфическим параметрам безопасности средств доступа этих первичных источников. Другими словами, необходимо разработать и реализовать унифицированную расширяемую модель управления правилами разграничения доступа. Кроме того, требуется обеспечить синхронизацию учетных данных пользователей («наследование» уровня доступа) в собственно ESB и подключенных к ней системах.
- Обеспечение выполнения любого процесса доступа к данным в контексте безопасности, связанным с пользователем, инициировавшим этот процесс независимо от того, является он локальным или удаленным по отношению к самому пользователю («виртуализация» пользователя).
- Программное шифрование входящего/исходящего трафика, ассоциированного с конкретным прикладным процессом на симметричных или несимметричных ключах на основе поддержки гетерогенной инфраструктуры хранения ключевой информации.