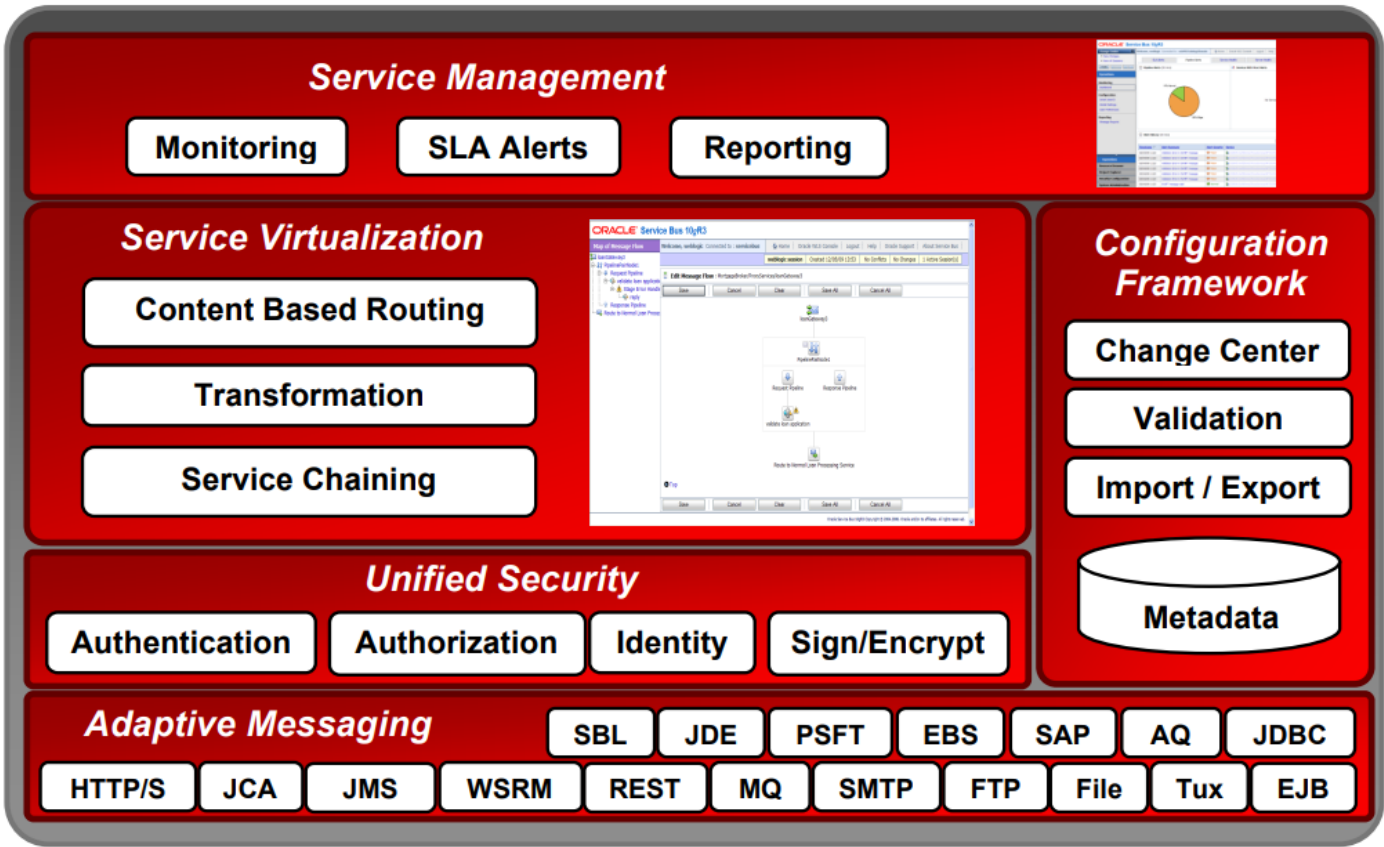


Давайте рассмотрим одну из популярных шин - [Oracle Bus](#).

Согласно официальной документации общее представление выглядит так:



Основные компоненты Oracle Service Bus (Oracle Bus) включают:

1. Прокси-сервисы (proxy-services): предоставляют интерфейс для взаимодействия с внешними системами. Могут выполнять трансформацию данных и маршрутизацию запросов.
2. Бизнес-сервисы (business services): предоставляют подключение к реальным бизнес-сервисам. Могут включать адаптеры для интеграции с конкретными приложениями.
3. Медиаторы (mediators): обеспечивают преобразование и трансформацию данных между различными форматами и протоколами.

4. Роутеры (routers): определяют, какие компоненты будут обрабатывать конкретные запросы на основе правил маршрутизации.
5. Очереди (queues): обеспечивают асинхронную передачу данных между компонентами.
6. Системы управления (management system): позволяют мониторить и управлять компонентами Oracle Bus.
7. Реестр сервисов (service registry): хранит метаданные о доступных сервисах для упрощения их обнаружения.
8. Трансформационные службы (transformation services): отвечают за изменение структуры данных в соответствии с заданными правилами.

Эти компоненты совместно обеспечивают интеграцию и обмен данными между различными системами в рамках Oracle Service Bus.

Обычно все ESB имеют интерфейс для настройки той или иной интеграции. То есть используют подход "low-code", когда вам не нужно писать код, а вы управляете некими уже заложенными сущностями и указываете для них параметры/протоколы/форматы сообщений. При покупке лицензии шины в предприятии обязательно должен быть выделен ресурс на изучение самой системы и на поддержку шины.

Мы кратко ознакомились с понятием ESB. После изучения раздела о брокерах сообщений вы, вероятно, осознали, насколько вопрос "что лучше, ESB или брокеры?" некорректен. Эти системы имеют существенные различия в архитектурных подходах и они служат разным целям:

- Брокеры сообщений — акцент на асинхронной передаче данных между разными сервисами, обеспечивая надежность и устойчивость. По сравнению с ESB стоимость запуска, поддержки и разработки ничтожно мала.
- ESB — более широкий спектр интеграционных функций, таких как медиация, трансформация, маршрутизация, что позволяет связать и согласовать различные приложения и сервисы. Стоимость запуска, поддержки и разработки может занять очень много ресурсов.