# 名词解释

## EAI(Enterprise Application Intergration)

**全称：**Enterprise Application Integration 企业应用集成

**用途：**集成基于各种不同平台、用不同方案建立的异构应用，包括使应用程序能够发送、接收信息，并对信息做出反应。

## MOM (Message Oriented Middleware)

**全称：**Message Oriented Middleware 基于消息的中间件

**定义：**指利用高效可靠的消息传递机制进行平台无关的数据交流和控制处理，并基于数据通信来进行分布式系统的集成。

## IIOP(Internet inter-ORB Protocol)

**全称：**Internet Inter-ORB Protocol 互联网内部对象请求代理协议

**定义：**一种通信协议，规定了客户和服务器ORBs间的通信机制。用来在CORBA对象请求代理之间交流。还提供了JAVA RMI 和CORBA的互操作能力。

## IDL(Interface Definition Language)

**全称：**Interface Definition language（接口定义语言）

**定义：**描述性语言，是CORBA规范的一部分，是跨平台开发的基础，提供一套通用的数据类型，用于描述接口，不定义实现，类似C中的头文件。

## ORB(Object Request Broker)

**全称：**Object Request Broker 对象请求代理

**定义：**它是 CORBA 的核心组件。ORB 提供了识别和定位对象、处理连接管理、传送数据和请求通信所需的框架结构。

## SOAP(Simple Object Access Protocol)

**全称：**Simple Object Access Protocol 简单对象访问协议

**定义：**SOAP是在松散的、分布式网络环境中使用XML交换结构化信息的一种简单协议，用来执行服务调用。

## XSLT(Extensible Stylesheet Language Transfomation)

扩展样式表转换语言，用于将一种XML文档转换为另外一种XML文档，或者其他类型的文档，比如HTML和XHTML。

## SOA（Service-Oriented Architecture）

**全称：**Service-Oriented Architecture 面向服务的体系结构

**定义：**SOA是一种程序组织架构，是设计原则。它把系统中的功能抽象成一个个服务，通过基于消息机制的分布式接口和调用协议相互作用，构成系统。

## WSDL(Web Services Description Language)

**全称：**Web Services Description Language 网络服务描述性语言

**定义：**用来描述Web服务和说明如何与Web服务通信。它是基于XML的组件描述，用来描述服务。

## ODBC（Open DataBase Connectivity）

**全称：**Open DataBase Connectivity 开放数据库互连

**定义：**它微软公司所提供的，当前被业界广泛接受的，主要使用SQL来实现的，一组对数据库访问的标准API。

## WSFL（Web Service Flow Language）

**全称：**Web Services Flow Language 网络服务流程语言

**定义：**Web Service间工作流描述，是一个描述商业过程的规范。将分散的、功能单一的Web服务组织成一个复杂的有机应用。

## UDDI

**全称：**Universal Description, Discovery and Integration 统一描述、发现和集成

**定义：**UDDI是一套基于Web的、分布式的、为Web服务提供的信息注册中心的实现标准规范，用来发布、查找服务。

# 问答题

## 应用集成的分类和示例。

1. 表示集成：软件用户界面。为原来基于终端的应用软件提供PC界面。提供一个由多组件合成的应用软件

（案例：为多个大型机应用程序提供统一的基于Java的界面）

1. 数据集成：直接访问软件创建、维护并存储的信息。多个信息源综合数据进行分析和决策。

（案例：用大型机和Oracle的可执行信息系统）

1. 功能集成：代码级的软件集成。能够解决前两种方法可解决的问题。要求新软件具有其他程序的功能。在集成中暗含工作流，确保应用间的事务完整性

（案例：把供应商的系统集成到采购系统中）

1. 业务流程集成
2. B2B集成

## 元数据

元数据：是描述、解释、定位一个信息资源的结构化的信息。元数经常被称作数据的数据。元数据是信息共享和交换的基础和前提，用于描述源数据集各种特征。

元数据主要有下列几个方面的作用：

1) 描述和发现资源； 2) 管理资源集合； 3) 保存数字化资源；

4) 提供数据互操作和数据转换方面的信息。

元数据标准

OMG、CORBA 、UML、XML

## xml文档，语法。

* 1. XML文件的第一行必须是声明该文件是XML文件以及所使用的规范版本。
  2. 根元素：在XML文件中有且只能够有一个根元素。所有的子元素必须嵌套在一个根元素中，必须有完整的起始和结束标签。
  3. 元素规则：名字中不能包含空格，名字不能以数字或标点符号开头，名字不能以任何大小写的xml开头，左尖括号（<）后不可以有空格，子元素如果内容为空可以缩写标签。
  4. 属性必须有用引号括起的值。
  5. 标记之间不得交叉。
  6. 控制标记、指令和属性名称等英文要区分大小写。

## DTD与Schema的差别

* 丰富的预定义数据类型
* 用户自定义数据类型
* 属性分组
* 开放的——原型可更新
* 支持命名空间
* 本身是一种xml文档。

## 应用集成与设计模式的关系。

应用集成与软件复用密切相关。继往开来的考虑。

设计模式：设计模式不仅可以帮助实现软件复用，还有一些特殊的设计模式可以作为应用集成的解决方案。

工厂方法模式：Web Service 的异构系统之间的方法调用。

代理模式：Spring等动态代理。

适配器模式：接口转换。第三方API集成；新旧系统集成。

## ODBC

它微软公司所提供的，当前被业界广泛接受的，主要使用SQL来实现的，一组对数据库访问的标准API。

**基本结构：**

1）应用程序：应用程序嵌有的SQL语句在运行时被转换为若干个动态连接库中的ODBC函数

2）驱动程序管理器：负责管理和调度驱动程序

3）驱动程序：是相应于某个数据源的ODBC函数执行码，存放于动态连接库，提供给应用程序调用。

4）数据源：提供的数据可以是RDBMS，也可以是OODBMS或各类文件形式

**工作流程：**

1）调用驱动程序管理器，把目标数据源对相应的驱动程序调入动态连接库；

2）根据SQL语句，调用动态连接库中若干个相应的ODBC函数；

3）执行ODBC函数，把SQL语句以字符串的形式传到数据源处；

4）数据源执行所收到的SQL语句，把结果返回应用程序。

## ETL（Extract TransForm Load）

ETL负责将分布的、异构数据源中的数据抽取到临时中间层后进行清洗、转换、集成，最后加载到数据仓库或数据集市中，成为联机分析处理、数据挖掘的基础。

**数据的抽取**

• 全量抽取和增量抽取——触发器；时间戳；快照

**数据的清洗**

• 数据格式不一致、数据输入错误、数据不完整

• 源数据和目标数据需要进行数据模式或语义映射的转换

• 在数据库中进行数据加工

**数据转换**

• 不一致数据转换：

• 数据粒度的转换：

• 商务规则的计算：

**数据装载**

• 最佳方法取决于所执行操作的类型以及需要装入多少数据

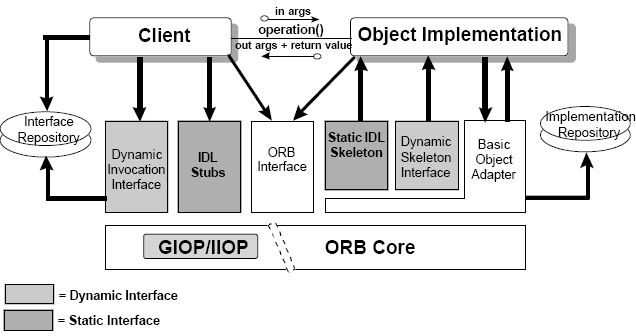
– 直接SQL语句操作，进行了日志记录并且可恢复

– 采用批量装载方法，易于使用，并且在装入大量数据时效率较高

**ETL的三种实现方法**

借助ETL工具—— SQL方式实现——两种结合。

## corba 2.0 体系结构图。描述功能（五个以上）



1、客户机应用程序：客户机应用程序用桩类型激发API或动态激发API向服务器发送请求

2、客户桩：一个客户机应用程序可以用客户桩给服务器应用程序发送请求。客户桩通过使用OMG IDL编译器编译OMG IDL接口而产生。

3、动态激发API:动态激发API是两种发送请求方式种的一种。动态激发API 在运行时从接口仓库里发现定义并用它来创建和激发向对象的请求。

4、ORB（客户）:ORB处理请求与相应服务器和方法的选择。当一个应用程序为某个对象上的某个操作向ORB发送一个请求时，ORB检验参数是否跟接口一致并把请求转发至服务器，如果需要的话启动服务器。

5、上下文对象:上下文对象包含有关客户机、环境的信息。上下文对象被表示成一列属性及其值。

6、接口仓库存储各个接口信息的模块，例如用OMG IDL编写的接口定义、常量、类型定义等，它们被当成定义的一部分使用。接口仓库包括对指定对象有效的操作的描述和对操作有效的参数描述。

7、ORB（服务器端）:ORB接收方法调度请求，激发服务器框架里的方法调度器，引导输出参数，并完成激发。

8、对象适配器：对象适配器处理与ORB相关的普通任务。

9、服务器框架：服务器框架提供调度请求到适当方法的必要代码。应用程序开发者通过编译接口定义得到服务器框架。

## 解释DOM和SAX解析XML文件的不同之处？

* + - 是事件驱动的流式解析技术，在解析XML文件时，不需要把整个XML文档加载到内存，不会建立对象模型、不支持随机访问、不支持数据更新、接口易用性较差、解析效率中等；
    - 是基于树形结构的对象XML解析技术，解析时整体装入和处理XML文档,系统资源占用大,效率低,速度慢，支持建立对象模型、支持随机访问、支持数据更新、接口易用性中等、解析效率较差。

## web service和SOA的关系

* + - Web服务是技术规范，而SOA是设计原则。特别是Web服务中的WSDL，是一个SOA配套的接口定义标准
    - SOA是一种架构模式，而Web服务是利用一组标准实现的服务
    - Web服务是实现SOA的方式之一

## 试分析消息中间件相对于分布式对象技术的优势

通讯程序可在不同的时间运行（传说中的异步传输）

对应用程序的结构没有约束（可靠性更强）