

Openbus - Integrador de dados e aplicações

19 de julho de 2011

1 O que é ?

O Openbus é uma arquitetura aberta e orientada a serviços (*Service-Oriented Architecture*, ou SOA) para integrar dados e aplicações heterogêneas de domínios distintos ou não. A arquitetura é baseada em um *barramento* em que os usuários se conectam para consumir ou prover serviços.

2 Por que utilizar o Openbus ?

Quando há necessidade de integração entre aplicações, algumas soluções podem ser apresentadas conforme a complexidade do ambiente a ser integrado. Caso o problema se resuma a troca de dados entre as aplicações, uma primeira solução é a integração externa, que consiste na exportação do dado em um determinado formato padronizado que por sua vez possa ser carregado pela aplicação de destino. Este processo é normalmente manual, deixando a cargo do usuário ou, do *job* que automatiza o processo, a responsabilidade pela sua execução. Um outro problema é a possibilidade de ocorrer perda de informação quando o formato padrão utilizado não consegue representar corretamente o dado de origem.

As opções de soluções disponíveis para a integração, podem ser ampliadas quando há a presença de aplicações de código aberto ou que ofereçam interfaces de programação. Para estes casos, uma integração programática pode ser realizada de forma direta, através de uma implementação que permita ao usuário comandar a transferência de dados. Este modo de interação agrega mais qualidade, diminuindo as chances de perda de informação durante a transferência do dado, pois os metadados das aplicações envolvidas podem ser considerados. Além da troca de dados, há a possibilidade de troca de mensagens entre as aplicações. O problema desta solução é que uma ponte deve ser desenvolvida para cada par de aplicações, o que resulta em um número quadrático de codificações de pontes em relação ao número de aplicações. Por exemplo, para integrar 10 aplicações precisamos implementar 45 pontes. Além disso, a troca de mensagens é naturalmente limitada entre cada par de aplicações.

Os problemas apresentados acima pela programação direta, podem ser eliminados com a adoção de uma plataforma comum de integração entre aplicações. Como por exemplo um *barramento*, em que cada aplicação se conecta a este, segundo um protocolo próprio deste barramento, e carrega ou exporta dados para

outras aplicações também conectadas ao barramento. Essa arquitetura define uma linguagem comum entre todas as aplicações, e, induz o compartilhamento de interfaces bem definidas para os serviços comuns entre as aplicações envolvidas, desta forma o esforço de codificação das pontes pode ser reduzido para um esforço linear, ou seja, para se integrar n aplicações, n pontes são necessárias. No mais, a troca de mensagens dinâmicas entre todas as aplicações passa a ser possível, possibilitando um ambiente de colaboração entre as aplicações.

3 Arquitetura

O Openbus é baseado no *middleware* CORBA. Seu barramento é formado por componentes que podem oferecer serviços ou podem apenas consultar serviços de outros componentes. Estes componentes são implementados através de um sistema de componentes para CORBA chamado SCS (Sistema de Componentes de Software). No SCS, os componentes são compostos por facetas que representam as interfaces que estes componentes oferecem.

A infraestrutura do Openbus provê dois serviços básicos necessários para que o usuário possa utilizar o barramento, o serviço de acesso e o serviço de registro. Um terceiro serviço auxilia o compartilhamento de informações entre um grupo de clientes, que é o serviço de sessão. Por último, a arquitetura do Openbus disponibiliza uma interface comum para a implementação de um serviço de dados.

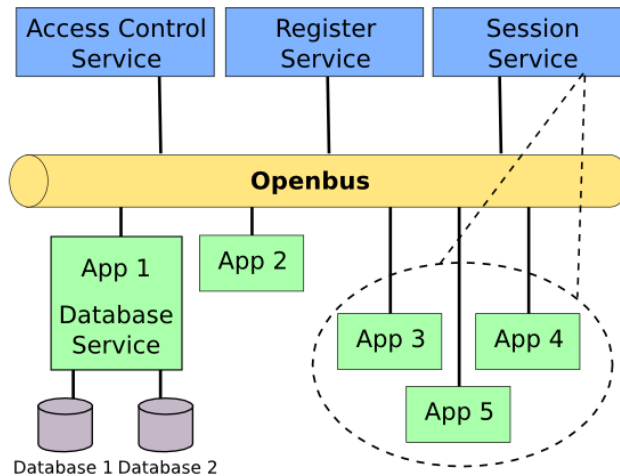


Figura 1: Arquitetura do openbus

Serviço de Acesso É o ponto de entrada no barramento. Sua referência é conhecida por todos. Para que um usuário possa entrar no barramento para consumir ou prover serviços, é necessário que ele se autentique no mesmo adquirindo uma credencial de acesso. A autenticação pode ser realizada através de um par usuário/senha ou através de um

certificado digital. Tipicamente, aplicações "clientes" se autenticam com um par usuário/senha, que normalmente será a identificação e a senha do usuário que iniciou a aplicação. Servidores tipicamente se autenticam através de um certificado.

Após a autenticação, uma credencial é emitida para o usuário para que o mesmo possa acessar os serviços disponibilizados pelo barramento. Essa credencial é composta por um identificador unívoco e por um nome de uma entidade à qual está associada, que pode ser um usuário ou um serviço. Essa credencial tem um ciclo de vida e deve ser renovada para que não expire.

Serviço de Registro Responsável por controlar as ofertas de serviços disponíveis no barramento. Um componente que queira oferecer um serviço deve explicitamente registrar sua oferta no serviço de registro. Componentes que desejam utilizar um serviço podem obter a localização e as propriedades de provedores desse serviço através de consultas ao serviço de registro.

Serviço de Sessão Oferece um mecanismo simplificado de troca de mensagens entre os componentes que compartilham uma mesma sessão.

Serviço de Dados Disponibilização de uma interface para a implementação de um serviço de acesso a dados hierárquicos ou não. A interface provê uma faceta para a navegação e outra para o acesso ao dado.

4 Como usar ?

O barramento é utilizado através de uma programação sobre uma biblioteca. Essa biblioteca é disponibilizada conforme a linguagem de programação que o usuário deseja desenvolver, e, conforme o ORB utilizado caso o usuário já utilize um em específico. As bibliotecas são oferecidas em *C++*, *Java* e *Lua*, e os ORB's contemplados são o *Orbix C++*, *JacORB*, *Mico* e *OiL*. Se nenhuma das bibliotecas ofertadas atenderem ao ambiente do usuário, o Openbus pode ser utilizado através das IDL's que são fornecidas para os serviços básicos.

Ainda não há um guia de programação para o usuário, mas temos um tutorial de programação para cada uma das linguagens suportadas oficialmente (*C++*, *Java* e *Lua*).

5 Plataformas suportadas

Algumas plataformas são suportadas oficialmente através de pacotes específicos que sofreram um processo de validação. Como o *software* é distribuído em código aberto, nada impede o usuário de realizar uma compilação em uma plataforma em específico que não é suportada oficialmente.

As plataformas suportadas são:

- Linux 2.6 x86 32bits GCC 4.x
- Linux 2.6 x86 64bits GCC 4.x

- Linux 2.6 x86 32bits GCC 3.x
- Linux 2.6 x86 64bits GCC 3.x
- Linux 2.4 x86 32bits GCC 3.x
- Linux 2.4 x86 64bits GCC 3.x
- Solaris 10 x86
- Solaris 10 sparc

6 Onde adquirir o pacote ?

O link do site oficial do OpenBus é:

<http://www.tecgraf.puc-rio.br/openbus>

Os pacotes do OpenBus podem ser adquiridos na [página de Download](#).

Há um pacote para cada plataforma que é suportada oficialmente. O pacote disponibiliza as bibliotecas referentes a cada uma das linguagens suportadas, e, o barramento em si.