

用 Oracle9IAS 及相关产品构建企业过程控制 HMI

Using oracle9ias forms and oracle develop enterprise process control hmi

方亮 康书广 王继军 (唐山钢铁股份有限公司自动化公司 河北唐山 063000)

摘要:本文主要介绍了如何用 Oracle9IAS、Forms 和 Oracle 为唐钢 1700mm 热轧薄板过程控制系统构建 Web 开发环境,具体包括 Oracle10g、Oracle9IAS、Forms 的安装、配置以及 Forms 应用程序在 Web 上的部署与发布,并比较完整地给出了开发企业过程控制人机交换界面(human machie interface)的方案。

关键词:Web Server Oracle Forms Oracle9IAS 过程控制 HMI

1 引言

随着人机交换界面(HMI)在企业中的广泛应用,一直使用的客户端/服务器模式构建建立的 HMI 平台以其管理维护复杂已经充分暴露了它的缺点。能不能不使用客户端/服务器模式而是通过瘦客户机即 WEB 浏览器向用户提供具有相同复杂度逻辑和用户所期望的接口的应用程序呢?如今随着基于 B/S(浏览器/服务器)模式的 Web 系统的大力发展,这一构想成为可能。下面我们就要介绍如何构建这样一个 B/S 结构的 Web 环境,并把用 Oracle Forms 开发出的 HMI 应用程序通过表单服务器部署在 Web 上。

2 相关软件介绍

完整的 Web 开发环境至少必须包含三个基本组件:Web 服务器、数据库以及服务器端后端技术。此外还有一些功能要求,例如可靠性、可伸缩性以及互操作性。为了满足上述条件,我选择使用 Oracle Forms、Oracle9ias、Oracle10g 作为搭建平台的应用软件,首先我们简单介绍一下这三种软件的背景及功能。

Oracle Forms 是用来开发可交互的、具有健壮数据校验功能的 GUI 界面的工具,使用 Oracle Forms 我们可以快速地开发一个能够自动管理通过 SQL 语言对数据库语言进行插入、更新、以及删除操作的应用程序。

Oracle9IAS 是一个能够及开发主页应用程序与发布站点的软件,能够发布基于 Java、Perl、Pl/sql、Xml、Forms 等编写的应用程序。它是一个与 Oracle Server

紧密集成的 HTTP 服务器,能够由存储在 Oracle 数据库的数据建立动态 HTML 文件。当数据改变时,这些 HTML 文件也自动更新,而不需要站点管理员的参与。这种方法能动态地实时地反映基于 Oracle 服务器的商务系统中的当前数据,而不是当今大多数站点上见到的静态的或不变的数据显示。通过 Oracle9ias,用户可以在其多层化的体系结构中部署基于 HTTP 和 HTML 的应用程序。中间层服务器集中管理应用程序逻辑,并将请求响应发回瘦客户机(通常情况下是 Web 浏览器)。原始数据库驻留在第三层,这样你就可以在该数据库上优化数据处理。

Oracle10g 是大家共知的数据库软件,它与 Oracle Internet 应用服务器(Oracle iAS)和 Oracle Internet 开发者套件(包括 Forms Developer)共同构成了 Oracle Internet 开发平台,该平台具有完整、简单、可靠、低成本等特点,能够帮助用户把数据管理、应用配置和开发等主要方面集中在一个平台上,能够有效地减少构建 IT 基础结构的成本和复杂性。

3 实际应用

3.1 实现原理

结合实际应用,我们使用客户端层、中间层、数据库层的三层体系结构来部署应用程序,客户端层包含 web 浏览器,浏览器是用来显示并使用应用程序的,中间层是应用程序服务器,在这一层保存了应用程序逻辑和服务器软件。数据库层是数据库服务器,在这一层保存了企业数据。当我们从客户端访问服务器程序

时,第一步我们需要从客户端浏览器上提交一个 URL,此动作触发了一个与 URL 相关的 GET 或 POST 请求方法操作。请求方法使用 HTTP 协议传给了 Web Listener。第二步 Web Listener 判断此 URL 是否与 Oracle Web Agent 有关(通过抽取 URL 的/owa 部分),并调用 Web Agent。第三步为连接到 Oracle Server, Web Agent 要求一些诸如连接到哪个服务器,使用什么用户名和口令等信息。这些信息作为 Web Agent 的一部分,存储在位于 OWS 管理目录下的文件 owa. cfg 中。第四步 Web Agent 通过解析环境变量 SCRIPT_NAME 来决定使用哪个服务,当连接到数据库后, Web Agent 调用相应的 PL/SQL 过程,其名字通过解析环境变量 PATH_INFO 获得。第五步如果 PL/SQL 过程要求参数,这些参数已经由 Web Listener 通过环境变量 QUERY_STRING 或标准输入(这依赖于请求方法是"GET"还是"POST")传给了 Web Agent。Web Agent 将把这些参数传给 PL/SQL 过程。在 Forms Developer 的帮助下, PL/SQL 过程从 Oracle10g 数据库中提取出数据并在 PL/SQL 表中生成了 HTML 文档。第六步如果 PL/SQL 过程

E:\oracle\product\10.1.0\Db_1,数据库名为 Prdb, 监听端口号为默认值 1521,机器名为 tstm0rm1。

(2) 安装 Oracle9 iAS 企业版本,安装前需要把安装 Oracle 10g 时候产生的系统环境变量由 PERL5LIB 改为 PERL5LIB_ORI, 否则安装 Oracle9IAS 时产生的 PERL5LIB 会与安装 Oracle 10g 时产生的 PERL5LIB 冲突,使安装崩溃。安装类型选择企业版,安装路径为 E:\oracle\ias,安装部件选择 Oracle9iAS Forms Server and Oracle9iAS Reports Server 及 Oracle HTTP Server in SSL Mode。Connect String 为服务器名:端口号:数据库名,本例中为 tstm0rm1:1521:prdb。

(3) 安装 Oracle Forms,安装名选择 Oracle_806,安装路径为 E:\oracle\06,安装类型为典型安装。此处只安装 Oracle Forms Developer 不安装 Forms Server For Testing,因为在安装 Oracle9IAS 的时候已经安装了 Oracle9iAS Forms Server and Oracle9iAS Reports Server。

3.2.2 软件的配置

软件的配置主要涉及到注册表的修改,控制文件的修改等,修改过程如下:

(1) 打开注册表 HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE,编辑子键 FORMS60_PATH,修改值数据如图 2。

(2) 编辑参数文件 E:\oracle\06\CONF\iserver.conf

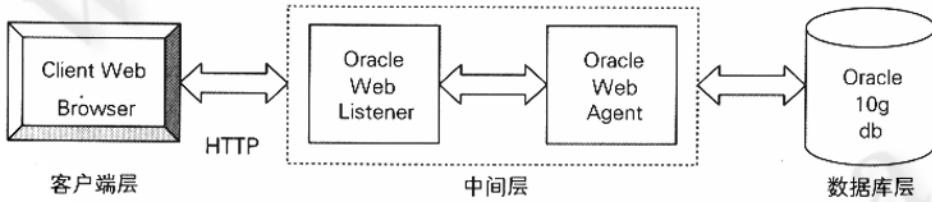


图 1

成功结束, Web Agent 将从 PL/SQL 表里取出生成的 HTML 文档,并把它发送到标准输出,那里是 Web Listener 期望 CGI 应用把他们的输出发送到的地方。如果 PL/SQL 过程返回错误, Web Agent 则把程序定义的出错页面发送到标准输出。 Web Listener 使用 HTTP 将 HTML 文档发还浏览器。浏览器并不知道请求的文档是在飞行中生成的,把它象任何其他 HTML 一样显示在屏幕上。原理图如图 1 所示。

3.2 相关软件的安装与配置

3.2.1 软件的安装

在安装软件时,需要先安装 Oracle,其次是 ias,最后是 Forms,它们的安装过程如下:

(1) 安装 Oracle 10g 数据库标准版,安装路径为

```

Alias /dev60html/ "E:\ORACLE\06\tools\web60\html/"
Alias /jinitiator/ "E:\ORACLE\06\JINIT/"
Alias /forms60java/ "E:\ORACLE\06\FORMS60\java/"
Alias /dev60temp/ "E:\ORACLE\806\tools\web60\temp/"
Alias /images/ "E:\ORACLE\06\web60\images/"
Alias /proj_icons/ "E:\na\tanghsm\src\Hmi\Oracle\Forms\OraRun\Icon/"
  
```

增加一行 Alias /proj_icons/ "E:\na\tanghsm\src\Hmi\Oracle\Forms\OraRun\Icon/",目的是为了指定程序在运行时所用到的 Icon 图标的路径。

(3) 编辑参数文件 E:\oracle\06\FORMS60\Server\formsweb.cfg 如下行,目的为指定 Web 的宽度、高度及样式。

```

; 3) Values for the Forms applet parameters:
width = 1100
height = 700
separateFrame = false
splashScreen = no
; select default background by not specifying a value
background =
lookAndFeel = Generic
    
```

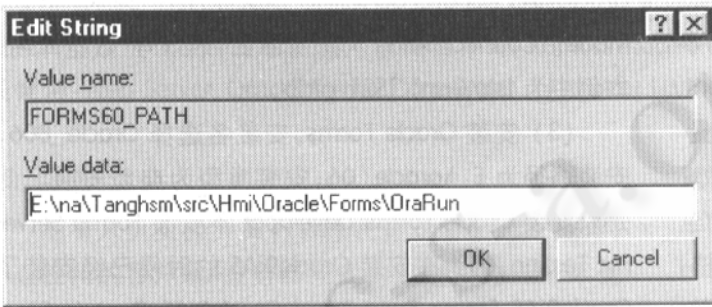


图 2

(4) 为了配置 Forms net 8 和 IAS net8, 需要分别编辑 E:\Oracle\06\NET80\ADMIN\tnsnames.ora 和 E:\Oracle\IAS\NETWORK\ADMIN\tnsnames.ora 添加如下行:

```

PRDB =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = TSTM0RM1)
    (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = prdb)
    )
  )
    
```

3.3 程序的部署与发布

当 Web 环境搭建完成后, 我们需要先把程序部署在 Web 上后, 才能通过浏览器访问它, 把应用程序通过 Forms Server 部署在 Web 上的步骤如下:

- (1) 把用 Forms 开发的应用程序全部编译, 生成 *.fmx 文件。
- (2) 把编译过的 *.fmx 文件拷贝到 E:\na\tanghsm\src\Hmi\Oracle\FORMS\OraRun 目录下, 也就是前面修改过的注册表子键 Forms60_Path 的路径。

(3) 在控制面板 -> 管理工具 -> 服务中分别启动 Oracle forms server 和 OracleiasHTTPServer 服务。

(4) 按如下方式编写 html 代码, 并保存为 html 格式的文件。

```

<HTML >
<HEAD > <TITLE > Start Oracle Forms </TITLE > </
HEAD >
<BODY >
<DD > <A href = http://tstm0rm1/dev60cgi/if-
cgi60.exe? config = ie50native&form =
primd&userid = prdb/a423pa@ prdb&par_ start-
type = CALL&par_ control = X&par_ display =
tstm0sb&par_lang = 0X0409&par_login = USER ~
9999999999999999 > Primary Data PRIMD
</A >
<DD > <A href = http://tstm0rm1/dev60cgi/if-
cgi60.exe? config = ie50native&form =
coils&userid = prdb/a423pa@ prdb&par_ starttype
= CALL&par_ control = X&par_ display = tstm0sb&par_
lang = 0X0409&par_login = USER ~ 9999999999999999
> Coil Schedule COILS </A >
<DD > <A href = http://tstm0rm1/dev60cgi/ifcgi60.
exe? config = ie50native&form = rolldfm&userid = prdb/
a423pa@ prdb&par_ starttype = CALL&par_ control =
X&par_ display = tstm0sb&par_lang = 0X0409&par_login
= USER ~ 9999999999999999 > Roll Data FM ROLLDFM
</A >
</BODY >
</HTML >
    
```

其中 http://tstm0rm1/dev60cgi/ifcgi60.exe? config = ie50native 为客户端提供 Web 服务的服务器地址, form = primd、form = coils、form = rolldfm、form = setptfm 中的 primd、coils、rolldfm、setptfm 分别为访问的程序名称。userid = prdb/a423pa@ prdb 为数据库的用户名、密码和数据库名。Primary Data PRIMD\Coil Schedule COILS\Roll Data FM 为 Html 文件执行后在网页上显示的 HMI 名称。

(5) 把生成的 html 文件拷贝到客户端。这样, 只要通过在客户端执行生成的 html 文件, 就能以 Web 的方式访问服务器程序了。

3.4 实际应用效果

如图 3.4 为客户端通过浏览器访问服务器上“粗轧设定画面”HMI 返回的结果,其中绿色部分的数据为原始数据,黄色部分的数据为模型计算结果数据,蓝色部分为可修改框。此 HMI 能把原始数据(如板坯厚

Server 操作系统、Oracle 10g 数据库平台上运行稳定。客户端通过浏览器访问服务器程序方便快捷,这样大大节省了部署应用程序时的管理和维护成本,使企业能够处于更有利的竞争地位。它结合了当今最先进的三层架构模式及数据存储技术,该技术的成功运用,

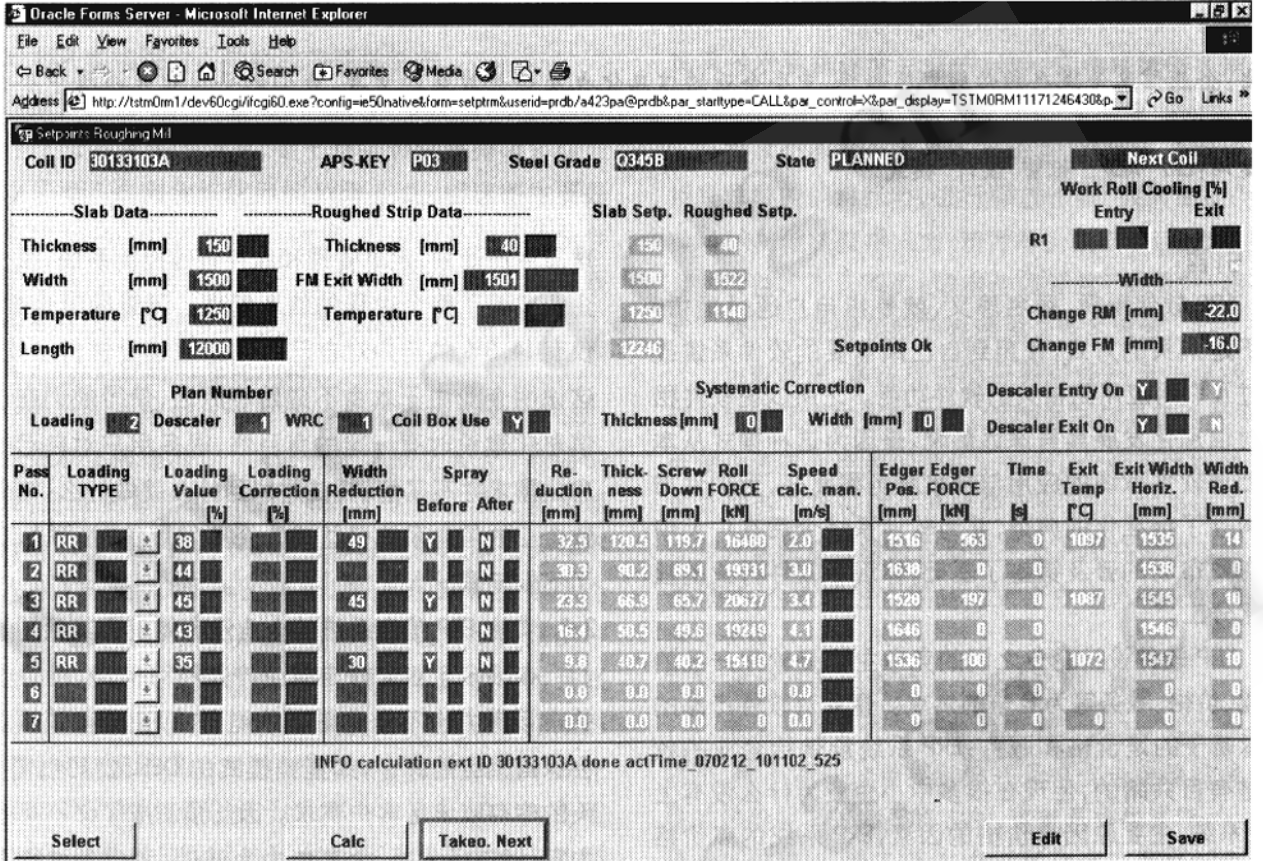


图 3

度、宽度、温度、长度等)和过程控制模型程序根据原始数据预计算出的结果(如压下量、出口厚度、辊缝位置、轧制力、轧制速度等)通过浏览器显示出来,这样操作人员任何一台操作终端上就可以实时监控查看要轧制板坯的预计算结果,并能对原始数据和计算偏差较大的数据进行修改、以此达到控制生产的目的。

必将成为企业在开发过程控制人机交换界面(HMI)上的优秀范例。

4 结束语

使用 Forms 开发,Oracle9IAS 提供部署与服务的应用程序在惠普 ML570 服务器、微软 Windows2000

参考文献

- 1 Oracle9i Web 开发指南/(美)布朗(Brown, B. D.)等著,武欣等译,机械工业出版社,2003.1.
- 2 Web Services 技术、架构和应用,电子工业出版社,2003.
- 3 Oracle 实用教程[M],北京电子工业出版社,2003.
- 4 Oracle 联机帮助
- 5 Siemens AG 技术附件