



AIDEE®

系统架构介绍，核心代码解析， 数据库结构介绍



日期: 2013/11/30
作者: 阳志军

目录



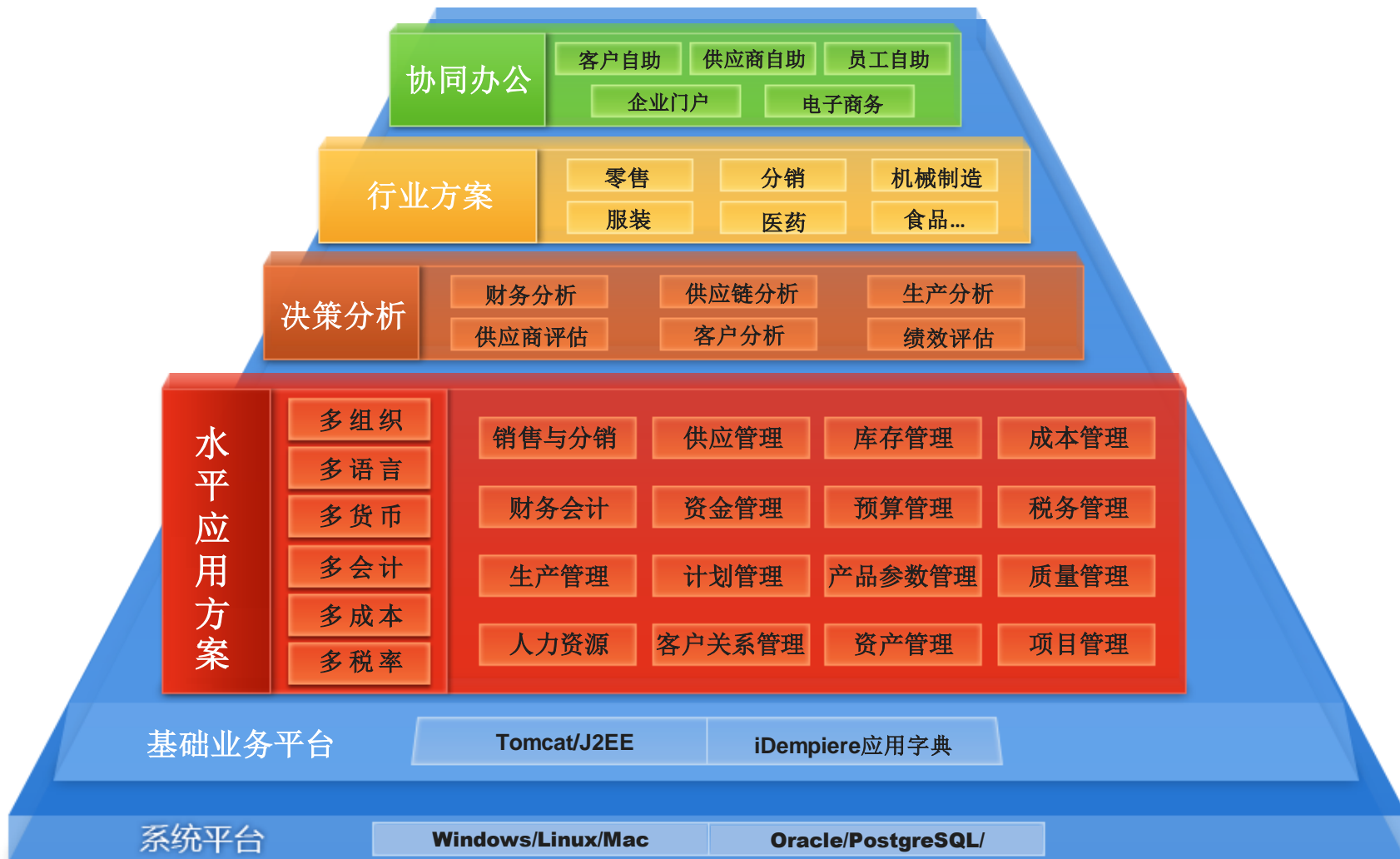
系统架构

核心代码解析

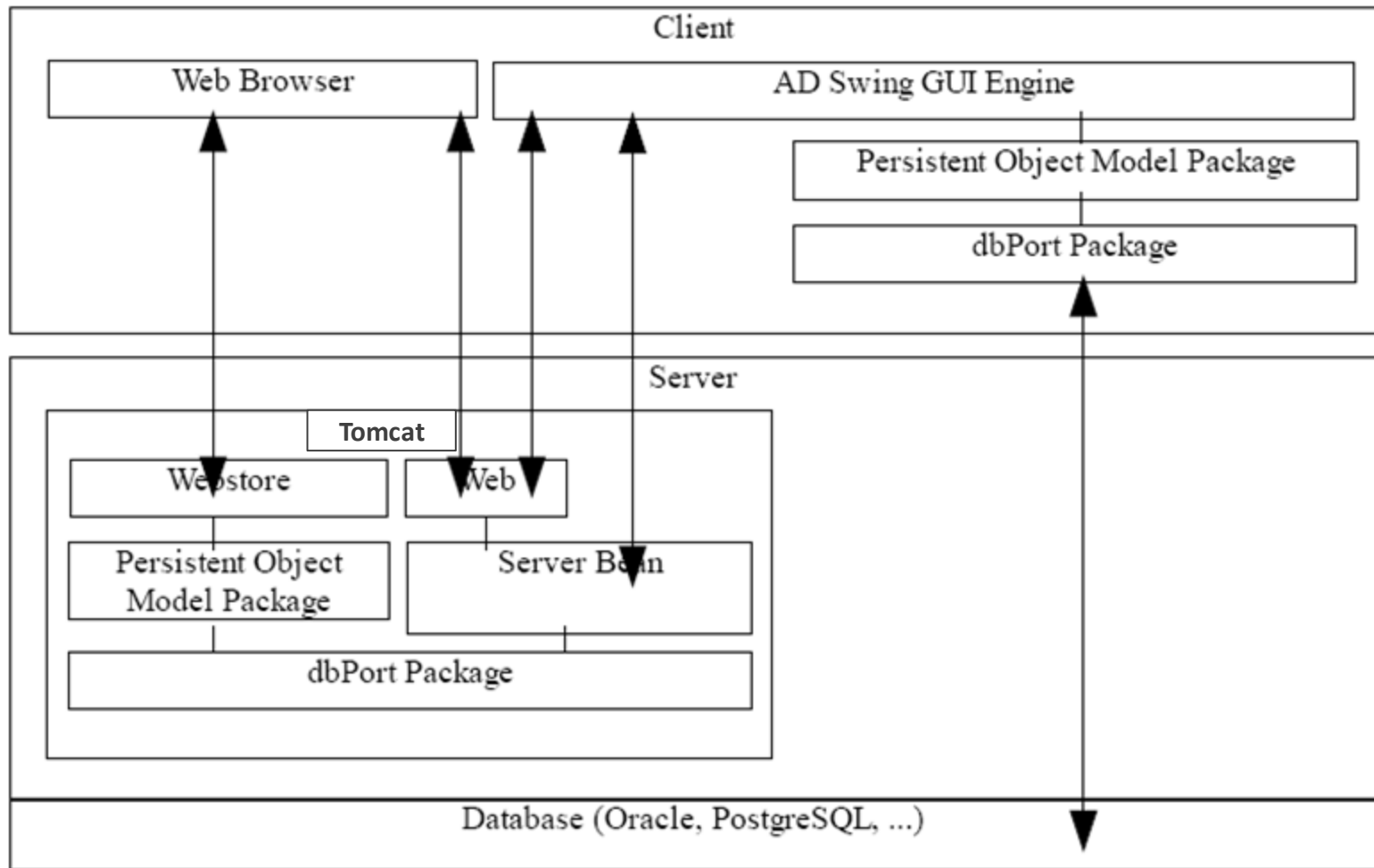
数据库结构介绍

Q&A

iDempiere产品架构

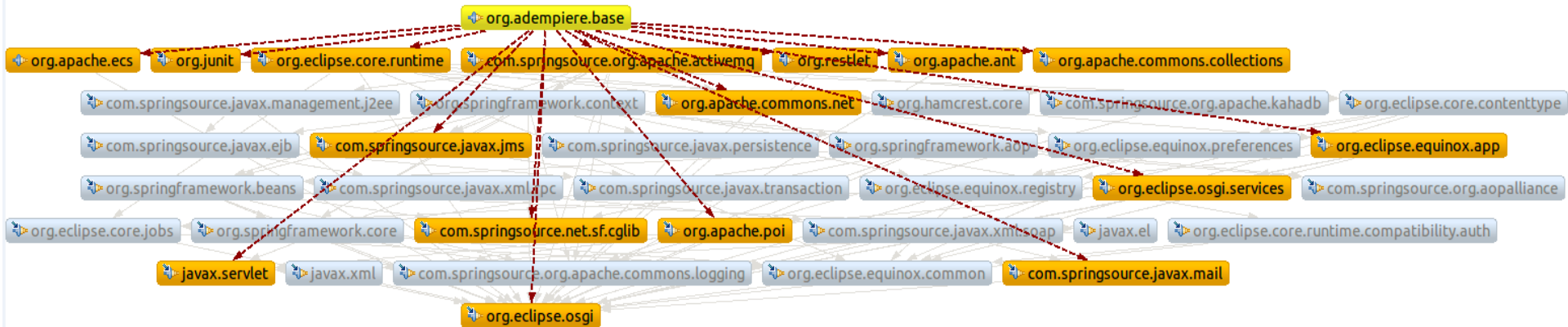


内部结构



OSGi Bundle 依赖关系图

org.adempiere.base



org.adempiere.server



目录



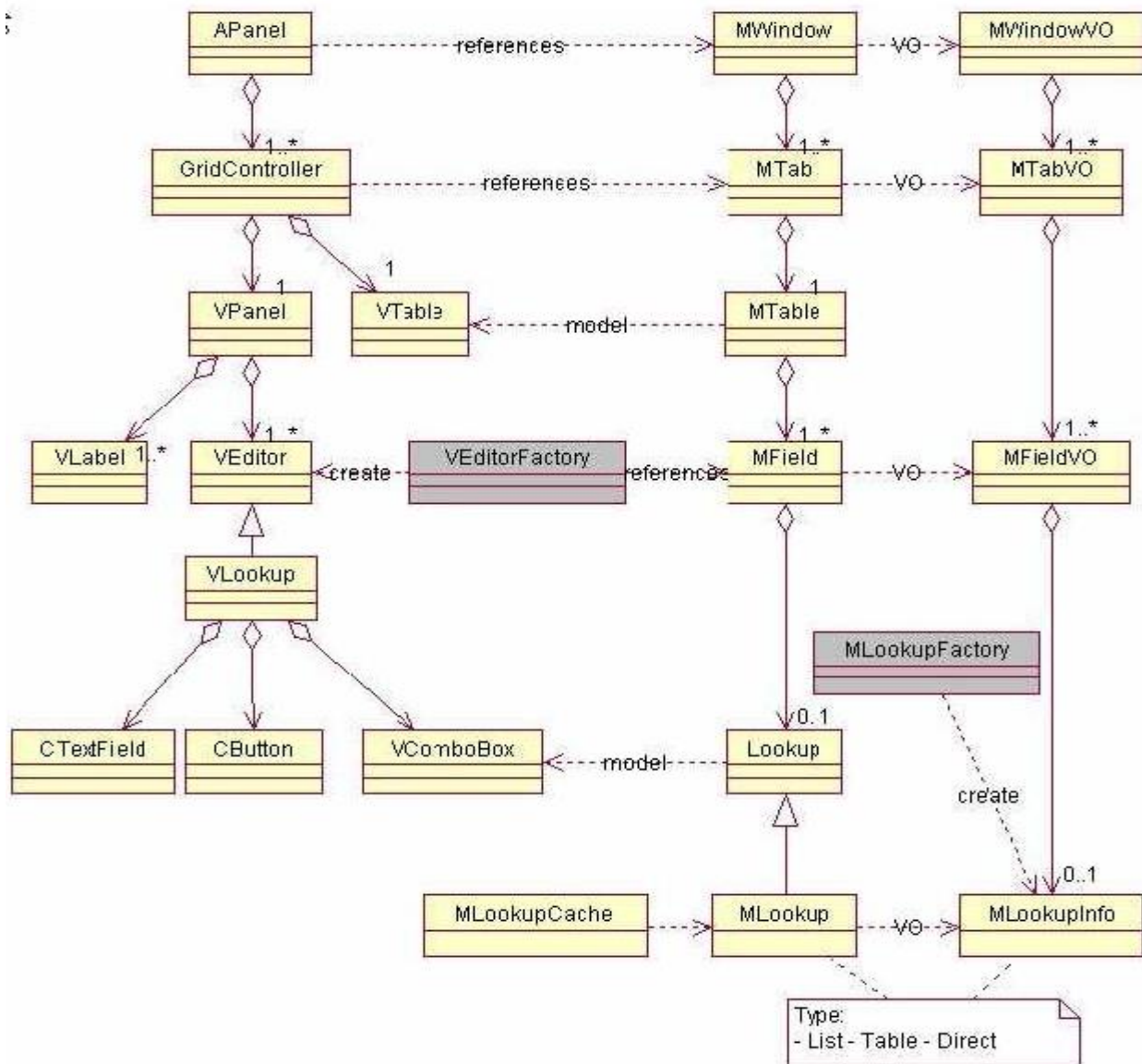
系统架构

核心代码解析

数据库结构介绍

Q&A

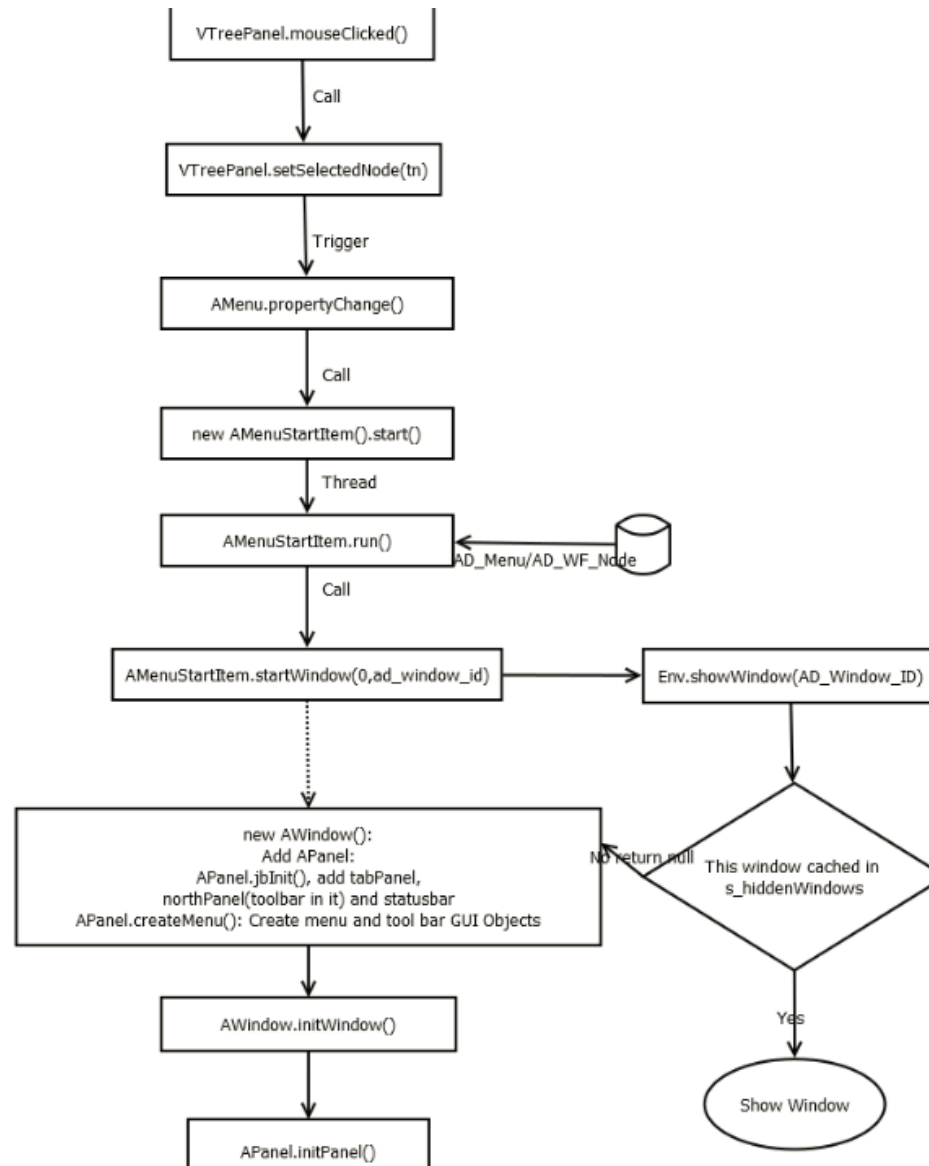
UI Model



Swing Window启动



PNG image



ZK Window启动

1. 启动WebUIServlet流程

`Bootstrap.main()` -> `Bootstrap.start()` -> `Catalina.start()` -> `Standard_XXX.start()` -> `StandardWrapper.loadServlet()` -> `WebUIServlet.init()` -> `DHtmlLayoutServlet.init(ServletConfig)` -> `new WebManager` -> `LanguageDefinition.init()` -> `DefinitionLoader.load()` -> `Classes.forNameByThread()` -> `AdempiereWebUI.<clinit>()`

2. webUIServlet响应浏览器请求生成Adempiere webui主页面的流程 (<http://localhost:8080/webui>)

`WebUIServlet.service()` -> `DHtmlLayoutServlet.doGet()` -> `DHtmlLayoutServlet.process()` -> `UiEngineImpl.execNewPage()` -> `UiEngineImpl.execCreate()` -> `UiEngineImpl.execCreateChild()` -> `SimpleUIFactory(AbstractFactory).newComponent()` -> `ComponentInfo.newInstance(Page, Component)` -> `ComponentDefinitionImpl.newInstance(Page, String)` -> `new AdempiereWebUI`

a. Adempiere webui 页面 onCreate 事件处理流程:

`EventProcessingThreadImpl.run()` -> `EventProcessingThreadImpl.process()` -> `EventProcessor.process()` -> `AdempiereWebUI.onCreate()` -> `Wlogin(AbstractUIPart).createPart()` -> `Wlogin.doCreatePart()` -> `new LoginWindow`

b. 修改zkwebui登陆界面: Wlogin.doCreatePart() -> String left = ThemeManager.getLoginLeftPanel() -> Theme.LOGIN_LEFT_PANEL_ZUL (login-left.zul).

i. 可以再doCreatePart注释掉left, 页面将不加载。

ii. 修改login-left.zul页面, 可以得到自己的左边布局

c. Change logo image on webui login page: loginPanel.java #145

Change right part, left part of Wlogin page: Wlogin.java #118, #101

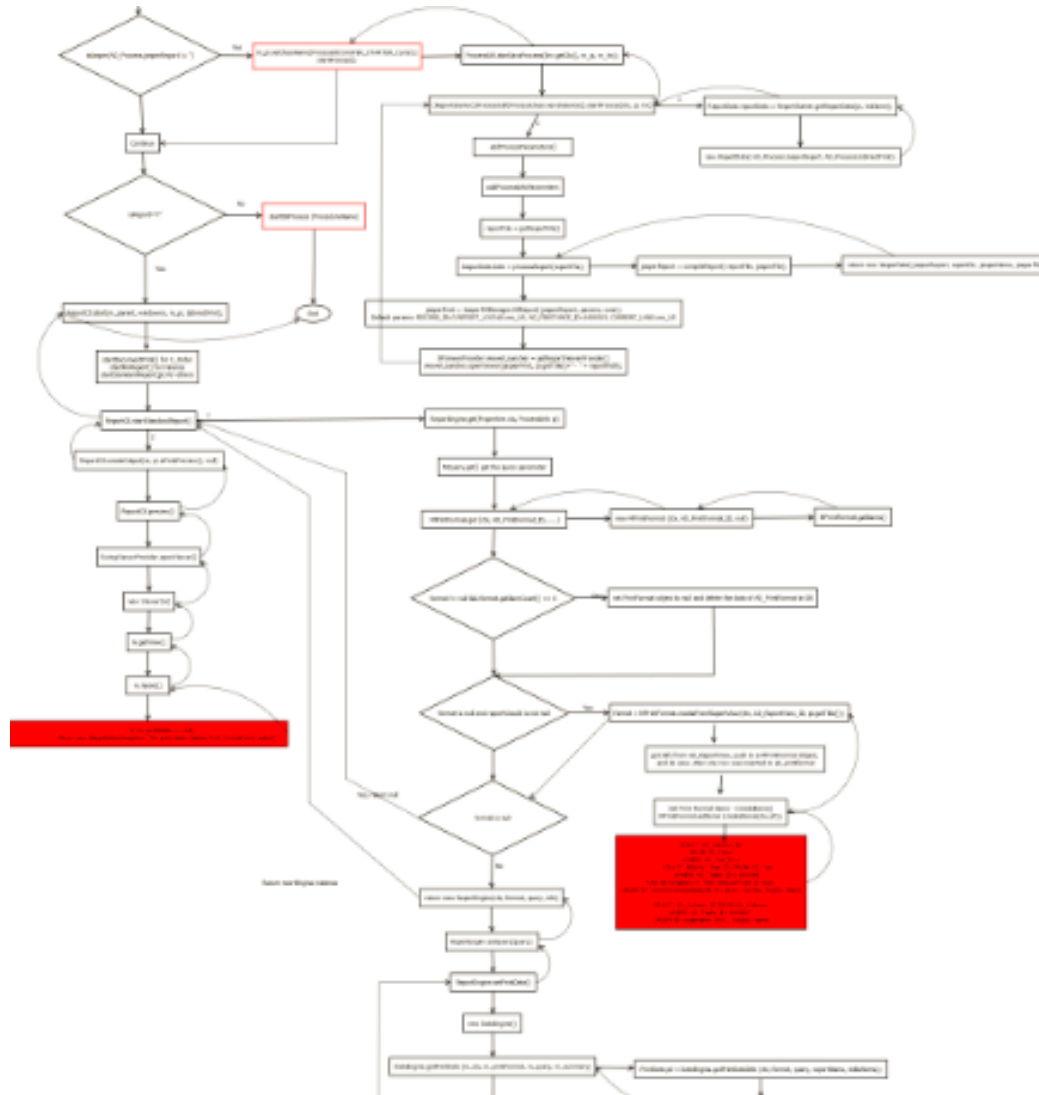
3. Zk webui的主界面生成逻辑代码:

- Menu, searchbar, toolbar初始化: `AdempiereWebUI.createDesktop()` -> `DefaultDesktop.doCreatePart()` -> `new SidePanel` -> `new MenuPanel` -> `MenuPanel.init()` -> `MenuPanel.initMenu(rootNode)` -> `MenuPanel.generateMenu(TreeChildren, rootNode)`
- HeaderPanel的初始化: `new HeaderPanel` -> `HeaderPanel.init()` -> `new West` -> `new Center`
- Home tab的初始化: `DefaultDesktop.createHomeTab()` -> `new Tabpanel()` -> `WindowContainer.addWindow(homeTab)` -> `MDashboardContent.getForSession()` -> `MDashboardContent.getZulFilePath()` -> `Executions.createComponents(URL)` -> `Panelchildren.appendChild(DashboardPanel)` -> `DashboarRunnable.addDashboardPanel()` -> `registerWindow(homeTab)` -> `Desktop.enableServerPush(true)` -> `DashboarRunnable.refreshDashboard()` -> `new Thread(DashboarRunnable)`
- 主界面: `new North` -> `HeaderPanel.setParent(North)` -> `new West` -> `SidePanel.setParent(West)` -> `new Center` -> `WindowContainer.createPart(Center)` -> `DefaultDesktop.createHomeTab()` -> `WindowContainer.addWindow(homeTab)`

4. Zk webui主界面的DashboardPanel是如何显示的, org.adempiere.webui.dashboard有8个DP*.java

- Favorites为例: `DPFavourites.java`
- `DefaultDesktop.doCreatePart()` -> `DashboardController.render()` -> `Executions.createComponents()` -> `new DPFavourites`
 - Favorites初始化在构造函数里做的: `DPFavourites()` -> `DPFavourites.createFavouritesPanel()`
 - 响应事件: `DPFavourites.onEvent()`
- `DefaultDesktop.java`里面定义了一个thread, 不断去更新HomeTab, `DefaultDesktop.renderHomeTab()`
 - `DefaultDesktop.updateUI()` -> `DefaultDesktop.renderHomeTab()` -> `DashboardController.render()`
 - `new MDashboardContent` -> `Anchorchildren.appendChild(Vlayout)` -> `Anchorlayout.appendChild(Anchorchildren)`
 - `new Panel` -> `Vlayout.appendChild(Panel)`
 - `new Panelchildren` -> `Panel.appendChild(Panelchildren)`
 - 然后在把内容装入PanelChildren, 可以分别装入HTML content, Window, Process&Report, Goal, 以及 ZUL File
 - zul文件有8个, 分别对应favorites, view, menu, menutree, recent items, activities, calendar, googlecalendar, performance
 - favourites的载入过程回到a)步骤

Process的调用过程



PNG image

Workflow调用

Workflow调用过程

1. workflow的实现相关的类

相关的几个class: `MWorkflow`, `MWFActivity`, `MWFProcess`, `WFNode`, `MWFNode`, `WFPanel`, `WFContentPanel`, `MWFNodeNext`, `WFLine`

2. 生成workflow页面:

1. `AMenuStartItem.run()` -> `AMenu.startWorkFlow(cmd)` -> `WFPanel.load(workflowID)` -> `Mworkflow.getNodes(ordered, clientID)` -> `Mworkflow.getNodesInOrder(clientID)` -> `Mworkflow.addNodesSF()`
2. `WFPanel.load(workflowID)` -> `WFContentPanel.add(WFNode, rw)` -> `WFContentPanel.add(WFLine, rw)`

3. 点击workflow页面的window或者是process:

1. `WFPanel.propertyChange()` -> `WFPanel.start(WFNode)` -> `AMenuStartItem(WFNodeID, notMenu, name, AMenu).start()`

4. 点击window窗口第一个tab里面的complete按钮, 比如purchase order里面的complete button, 触发workflow流程(由workflow process来执行Document Action):

1. `Apanel.actionPerformed()` -> `Apanel.actionButton()` -> `VDocAction(window, tab, Vbutton, recordID)` -> `VDocAction.actionPerformed()` -> `ProcessModalDialog.actionPerformed()` -> `ProcessCtl.process(processParameter, ProcessInfo)` -> `ProcessCtl.run()` ->
2. `ProcessCtl.run()` -> `ProcessCtl.startWorkflow(ctx, ProcessInfo, int)` -> `Mworkflow.start(ProcessInfo)` -> `MWFProcess.startWork()` -> `MWFActivity.run()` -> `MWFActivity.setWFState(WFState)` -> `MWFProcess.checkActivities()` -> `MWFProcess.startNext()` -> `MWFActivity.run()` -> `MWFActivity.performWork()` -> `BusinessObject.processIT(docAction)` -> `DocumentEngine.processIt(docAction)` -> `BusinessObject.xxxIt()`

5. Workflow如何被自动触发? 设置类型为"document process", 当business object对应的table值发生改变时, table对应的workload就被执行

1. 如process payment workflow, 对应的table是C Payment, 可以在application dictionary里面查到c_payment对应的workflow, 表列名为"DocAction"对应的process项值。当点击Payment窗口的complete按钮时, process payment工作流就会执行

6. workflow approve process:

`MWFActivity.setUserChoice()` -> if `MWFActivity.getNode().isUserApproval()` -> if `MWFActivity.isInvoker()` -> `MWFActivity.getResponsible()` -> `Mwosstimestampcard.processIt(DocAction.ACTION_Approve)` -> `Mwosstimestampcard.saveEx()` -> `MWFProcess.checkActivities()`

7. workflow审批的调用过程: 在主界面Workflow Activities点击ok按钮

1. `AMenu.dispatchEventImpl()` -> `Cbutton().dispatchEventImpl()` -> `Cbutton.processMouseEvent()` -> `WFActivity.actionPerformed()` -> `WFActivity.cmd_OK()` -> `MWFActivity.setUserChoice()` -> `MWFActivity.setWFState()` -> `MWFProcess.checkActivities()` -> `MWFActivity.setVariable()` -> `MWFActivity.setWFState()`
2. `MWFActivity.setUserChoice()` -> if `MWFActivity.isUserApproval()` is false -> newState = StateEngine.STATE_Aborted -> doc.processIt (DocAction.ACTION_Reject)
3. `MWFActivity.setUserChoice()` -> if `MWFActivity.isUserApproval()` is true -> if `isInvoker()`, then `MWFActivity.getApprovalUser()` -> doc.processIt (DocAction.ACTION_Approve) ->
4. 还存在一种不需要审批, 但需要确认, 比如Verify completed requisition, 流程如下:
`WFActivity.cmd_OK()` -> `MWFActivity.setUserConfirmation()`

8. main app的workflow activity里面显示的pending approve的activity是如何从数据库读数据的?

初始化`AMenu`, 准备`wfActivity` Panel: `wfActivity = new WFActivity(this)` -> `WFActivity.dynInit()` -> `WFActivity.loadActivity()` -> SQL to fetch data from table: AD_WF_Activity

目录



系统架构

核心代码解析

数据库结构介绍

Q&A

iDempiere数据表介绍

<http://www.adempiere.com/schemaspy/>



目录

版本更新记录.....	2
Compiere数据表介绍	4
第一个列: ad_client_id.....	4
第二个列: ad_org_id	4
第三个列: isactive	5
第四个列: created	5
第五个列: createdby	5
第六个列: updated	5
第七个列: updatedby	5
Compiere数据表的设计规则是:	6

iDempiere数据库编程模版

- PostgreSQL大小写规则:

列名, 表名等本身默认就是大小写不敏感. PostgreSQL内部全部转成小写.

如果是值的话, 比如英文(只有英文吧), 区分大小写(当然的).

你只能用lowerCase和upperCase转换. 需要注意这样的话, 索引基本上是废掉了. 当然可以建立函数索引等..

- 模版代码一:

```
String sql = "SELECT u.c_bpartner_id FROM ad_user u WHERE u.ad_user_id = ?";
Env.setContext(Env.getCtx(), "#C_BPartner_ID", DB.getSQLValue(null, sql, C_BPartner_ID));
```

- 模版代码二:

```
String whereClauseFinal = "C_Billing_ID=? ";
List<MCBillingLine> list = new Query(getCtx(), MCBillingLine.Table_Name, whereClauseFinal, get_TrxName())
    .setParameters(new Object[]{getC_Billing_ID()})
    .setOrderBy(MCBillingLine.COLUMNNAME_Line)
    .list();

return list.toArray(new MCBillingLine[list.size()]);
```

- 模版代码三:

```
PreparedStatement pstmt = null;
ResultSet rs = null;
try
{
    pstmt = DB.prepareStatement(sql, null);
    pstmt.setString(1, locatorValue.trim());
    rs = pstmt.executeQuery();
    while (rs.next())
    {
        //m_locator_id
        int M_Locator_ID = rs.getInt("M_Locator_ID");
        if (M_Locator_ID != 0)
            locator_ID = M_Locator_ID;
    }
}
catch (SQLException e)
{
    log.log(Level.SEVERE, "locator", e);
}
finally
{
    DB.close(rs, pstmt);
    rs = null; pstmt = null;
}
```

52个存储过程介绍

Function	Owner
acctbalance(numeric, numeric, numeric)	adempiere
add_months(timestamp with time zone, numeric)	adempiere
adddays(timestamp with time zone, numeric)	adempiere
adddays(interval, numeric)	adempiere
altercolumn(name, name, name, character varying, character varying)	adempiere
bompricelimit(numeric, numeric)	adempiere
bompricelist(numeric, numeric)	adempiere
bompricestd(numeric, numeric)	adempiere
bomqtyavailable(numeric, numeric, numeric)	adempiere
bomqtyavailableasi(numeric, numeric, numeric, numeric)	adempiere
bomqtyonhand(numeric, numeric, numeric)	adempiere
bomqtyonhandasi(numeric, numeric, numeric, numeric)	adempiere
bomqtyordered(numeric, numeric, numeric)	adempiere
bomqtyorderedasi(numeric, numeric, numeric, numeric)	adempiere
bomqtyreserved(numeric, numeric, numeric)	adempiere
bomqtyreservedasi(numeric, numeric, numeric, numeric)	adempiere
bpartnerremitlocation(numeric)	adempiere
charat(character varying, integer)	adempiere
currencybase(numeric, numeric, timestamp with time zone, numeric, numeric)	adempiere
currencyconvert(numeric, numeric, numeric, timestamp with time zone, numeric, numeric)	adempiere
currencyrate(numeric, numeric, timestamp with time zone, numeric, numeric, numeric)	adempiere
currencyround(numeric, numeric, character varying)	adempiere
daysbetween(timestamp with time zone, timestamp with time zone)	adempiere
documentno(numeric)	adempiere
firstof(timestamp with time zone, character varying)	adempiere
get_sysconfig(character varying, character varying, numeric, numeric)	adempiere
getdate()	adempiere
instr(character varying, character varying)	adempiere
instr(character varying, character varying, integer)	adempiere
instr(character varying, character varying, integer, integer)	adempiere
invoicediscount(numeric, timestamp with time zone, numeric)	adempiere
invoiceopen(numeric, numeric)	adempiere
invoiceopentodate(numeric, numeric, timestamp with time zone)	adempiere
invoiceopentodate(numeric, numeric, date)	adempiere
invoicepaid(numeric, numeric, numeric)	adempiere
invoicepaidtodate(numeric, numeric, numeric, timestamp with time zone)	adempiere
nextbusinessday(timestamp with time zone, numeric)	adempiere
nextid(integer, character varying)	adempiere
nextidbyyear(numeric, numeric, character varying)	adempiere
nextidfunc(integer, character varying)	adempiere
paymentallocated(numeric, numeric)	adempiere
paymentavailable(numeric)	adempiere
paymenttermdiscount(numeric, numeric, numeric, timestamp with time zone, timestamp with time zone)	adempiere
paymenttermduedate(numeric, timestamp with time zone)	adempiere
paymenttermduedays(numeric, timestamp with time zone, timestamp with time zone)	adempiere
productattribute(numeric)	adempiere
round(numeric, numeric)	adempiere
subtractdays(timestamp with time zone, numeric)	adempiere
subtractdays(interval, numeric)	adempiere
trunc(timestamp with time zone)	adempiere
trunc(interval)	adempiere
trunc(timestamp with time zone, character varying)	adempiere

目录



系统架构

核心代码解析

数据库结构介绍

Q&A

谢谢！

欢迎与我联系：

网址：www.aidee.cn

邮箱：ayang@aidee.cn

