

ЛОЦМАН:PLM

Версия 8

Расширение возможностей системы.

Дополнительные модули.

Управление меню.



Способы расширения возможностей системы

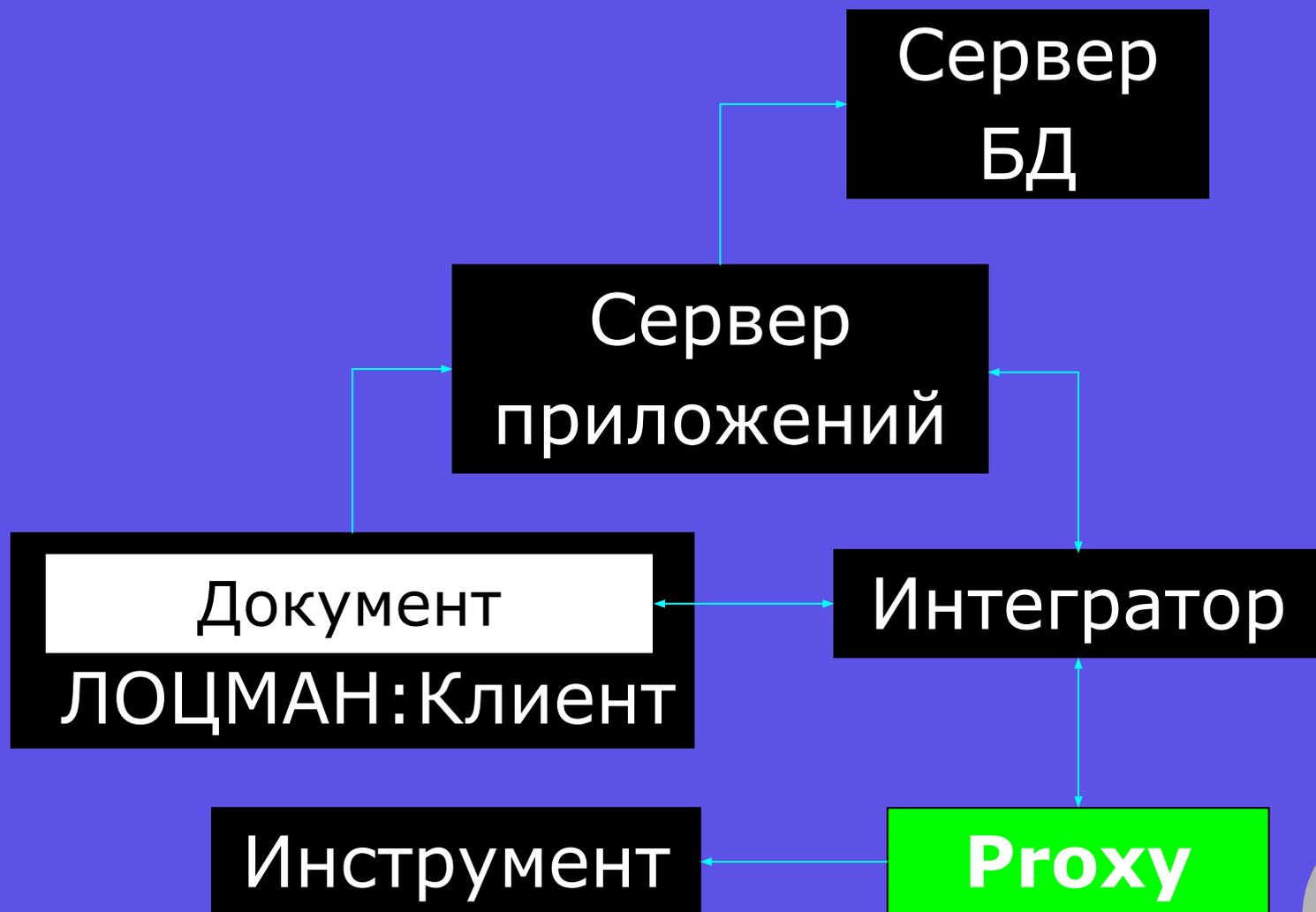
- **Плагин** – внешняя библиотека расширяющая функционал клиента.
- **Прокси** – библиотека интеграции с инструментом
- **Специальные клиенты** – самостоятельные приложения использующие методы СП для манипуляций с объектами БД
- **Бизнес-объект** – внешний справочник - поставщик данных для системы



Взаимодействие компонентов системы и дополнительных модулей (плагинов)



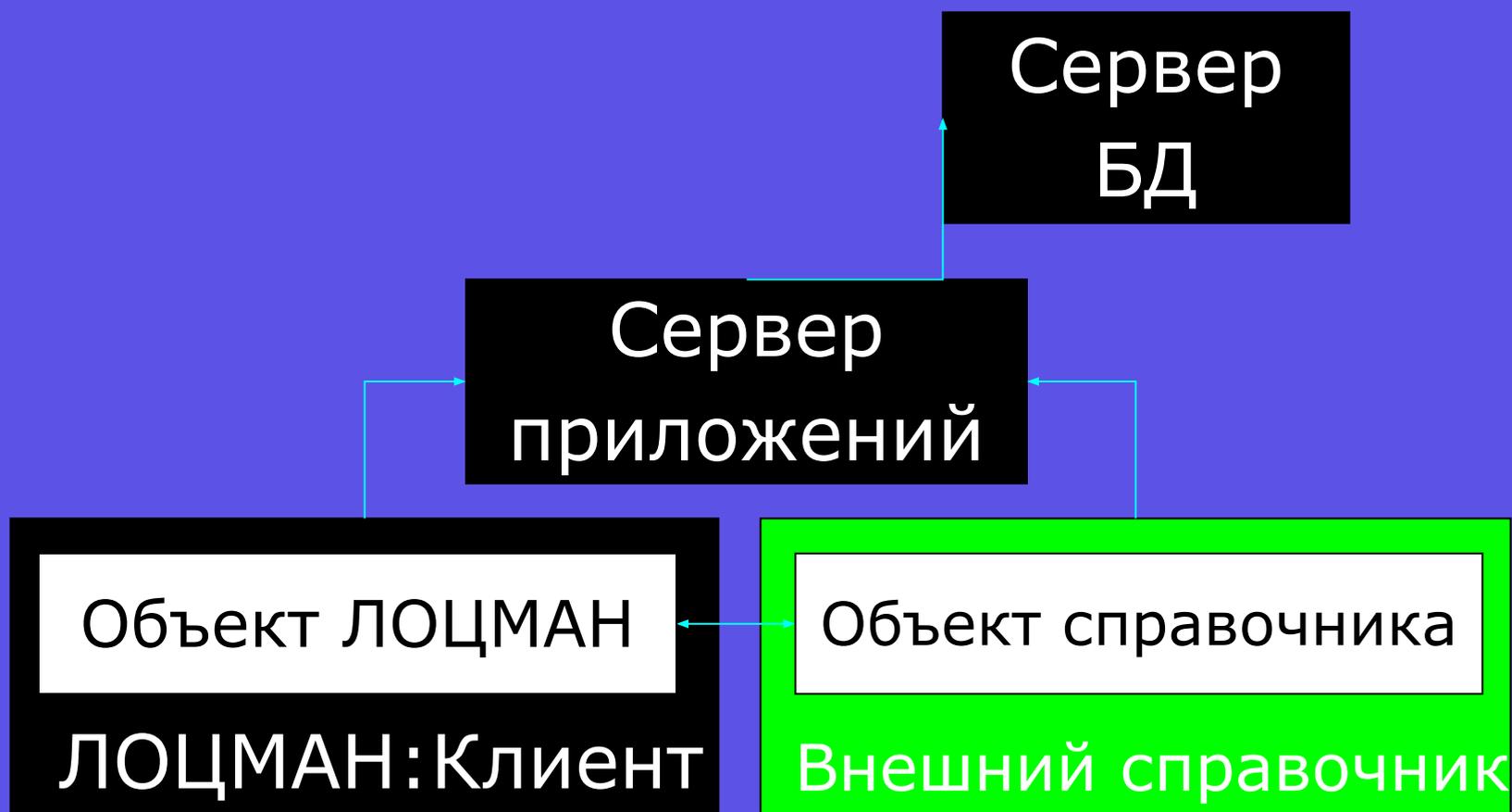
Взаимодействие компонентов системы и библиотек интеграции



Взаимодействие компонентов системы и специального клиента



Взаимодействие компонентов системы и бизнес-объектов



Плагин это ...

Динамически подключаемая библиотека (DLL), с расширением «*.pgi».

Плагин должен содержать 2 обязательные функции, имеющие следующее назначение:

InitUserDLL – построение структуры меню;

PgiCheckMenuItem – контроль доступности пунктов меню.

... и набор процедур-обработчиков, соответствующих пунктам меню



Управление меню

Индикаторы меню и функции меню помещаются в меню-модуль

Возможные положения пунктов меню

```
MI_FILE, AFTER_MI_FILE, BEFORE_MI_FILE,  
MI_VIEW, AFTER_MI_VIEW, BEFORE_MI_VIEW,  
MI_EDIT, AFTER_MI_EDIT, BEFORE_MI_EDIT,  
MI_TOOLS, AFTER_MI_TOOLS, BEFORE_MI_TOOLS,  
MI_HELP, AFTER_MI_HELP, BEFORE_MI_HELP
```

«Инструменты», приставки **BEFORE_** и **AFTER_** указывают, размещение, соответственно, перед или после пункта меню

stFunction – имя процедуры-обработчика



InitUserDLL

Вызывается дважды. Первый раз - для определения объема и выделения памяти под новые пункты меню. Второй – для заполнения выделенной памяти информацией о структуре меню и процедурах-обработчиках, соответствующих пунктам меню.

```
function InitUserDLL (Value : Pointer) : Integer;
  stdcall; export;
var
  item : PAddMenu;
begin
  if Value = nil then //первый вызов
    Result := 2 //сколько будет пунктов в меню
  else begin          //второй вызов
    item := Value;
    item.stMenu := 'AFTER_MI_VIEW#Плагин#Функция №1';
    item.stFunction := 'menu_Func1';
    inc(item);
    item.stMenu := 'AFTER_MI_VIEW#Плагин#Функция №2';
    item.stFunction := 'menu_Func2';
  end;
end;
```



PgiCheckMenuItem

Вызывается при каждом обращении пользователя к меню. Позволяет, в зависимости от параметров выбранного к дереву клиента объекта и других условий, устанавливать активность пунктов меню

```
function PgiCheckMenuItem(stFunction: string;
  RunMethod: TRunMethod;
  ParentVersion, ChildVersion: TVersion;
  stLinkName, DataBaseName, stCheckOut: string):
  boolean; stdcall;
begin
  AssignRunMethod(RunMethod);
  Result := False;
  if stFunction = 'menu_Func1' then //ТОЛЬКО ДОКУМЕНТЫ
    Result := ChildVersion.boDocument;
  if stFunction = 'menu_Func2' then //ТОЛЬКО ОБЪЕКТЫ
    Result := not ChildVersion.boDocument;
end;
```



Класс TVersion

Предназначен для описания объектов базы данных. Реализация находится в PIClasses.pas.

Содержит основные идентификационные данные объекта:

inID – Уникальный идентификатор,

stType – Тип объекта,

stProduct – Ключевой атрибут

stVersion – Номер версии

boDocument – Признак документа

Имеет несколько конструкторов

FromEmpty - создает пустой объект;

FromRecord - из набора данных TDataSet

FromIDVersion - по ID объекта



```
library PgiExample;

uses
  SysUtils, Classes,
  PIClasses in 'PIClasses.pas',
  Unit1 in 'Unit1.pas',
  frmInsertObjectUnit in 'frmInsertObjectUnit.pas',
  frmNewVersionUnit in 'frmNewVersionUnit.pas';

{$E pgi}

{$R *.res}

exports
  InitUserDLL, PgiCheckMenuItem,
  menu_GetAttrList, menu_InsertObject, menu_NewVersion;

begin
end;
```



**инициализация, проверка меню и
процедуры-обработчики должны
быть объявлены в секции exports**



Вызовы методов сервера приложений

Доступ к методам СП обеспечивается посредством функции **RunMethod**, которая реализована в клиентском модуле и используется им для обращения к СП. Указатель на **RunMethod** передается в плагин как аргумент функции **PgiCheckMenuItem** и процедур-обработчиков.

```
//Объявление прототипа функции RunMethod
TRunMethod = function(
    stName: string;
    Params: array of Variant): Variant of object;
```



Управление клиентом

ЛОЦМАН клиент поддерживает набор сообщений для обновления данных, перемещения по дереву объектов и других операций:

- WM_REFRESHVERSION = WM_USER + 1;
- WM_REFRESHPARENT = WM_USER + 4;
- WM_GOTOCHILD = WM_USER + 5;
- WM_REFRESHCHECKOUTLIST = WM_USER + 6;
- WM_REFRESHPROJECTLIST = WM_USER + 7;
- WM_GOTONODE = WM_USER + 8;
- WM_GOTOOBJECT = WM_USER + 9;
- WM_OPENOBJECTINNEWWINDOW = WM_USER + 100;
- WM_OPENOBJECTSINNEWWINDOW = WM_USER + 101;

```
//обновить объект с идентификатором inId  
PostMessage (ClientHandle,  
WM_REFRESHVERSION, inId, 0) ;
```



Подключение плагина

Плагины подключаются к базам данным в секции Plugins файла

... \Client\Loodsman.ini,

...

[Plugins]

Имя_БД=плагин1.pgi;плагин2.pgi;...;плагинN.pgi

[AppServers]

ServerList=AK-20

[FileTemplates]

Документы Microsoft Word=C:\Loodsman7\Client\Документы
Microsoft Word

[Plugins]

СПДС=LoodsmanSpds.pgi;SPDSAcadDex.pgi;PgiExample.pgi

ИЯ

