

Разработка прикладного программного обеспечения интегрированных информационных систем управления на основе UML.

Гогичаишвили Георгий Георгиевич¹, Сургуладзе Гия Георгиевич²

^{1,2} кафедрой "Автоматизированных систем управления" Грузинского Технического Университета.

Аннотация

предлагается концепция и технология разработки интегрированных информационных систем управления (ИИСУ) организационными и производственными предприятиями на базе UML-технологии и клиент-серверной архитектуры. Рассматриваются задачи анализа, моделирования и визуально-компонентного программирования мультипроцессорных систем на базе стохастических сетей Петри и объектно-ориентированного подхода, с использованием инструмента Borland C++Builder.

Ключевые слова:

интегрированные информационные системы управления, унифицированный язык моделирования (UML), распределенная база данных, стохастические сети Петри, интерфейсные программные средства, визуальное и компонентное программирование.

Начало XXI века характеризуется бурным развитием компьютерной индустрии и информационных технологий. Появились новые, эффективные и мобильные аппаратные и телекоммуникационные средства связи, передачи и обработки информации. На совершенно новом уровне проектируются и создаются глобальные и локальные компьютерные сети. Вследствие этого появились и быстро развиваются новые информационные технологии с высококачественными пакетами универсальных, стандартных и прикладных программ.

В первую очередь следует упомянуть методологию UML (Unified Modeling Language) технологии, которая, опираясь на объектно-ориентированный и структурный подходы к моделированию, наследственно объединила в себе такие важные научные направления теоретической и практической информатики, какими являются: автоматизация проектирования, автоматизация программирования и автоматизация обработки баз данных [1]. UML-технология справедливо принята в качестве международного стандарта по созданию прикладного программного обеспечения компьютерных систем [2].

В настоящей работе предлагается концепция и технология разработки интегрированных информационных систем управления (ИИСУ) организационными и производственными предприятиями на базе UML-технологии и клиент-серверной архитектуры.

На рис.1 представлена обобщенная схема типичной «человеко-машинной» ИИСУ. Предметом исследования, в этом контексте, является множество структурных компонентов (проблемы, интерфейсы, пакеты программ) системы и их взаимосвязь на базе преобразовательных коммуникационных средств.

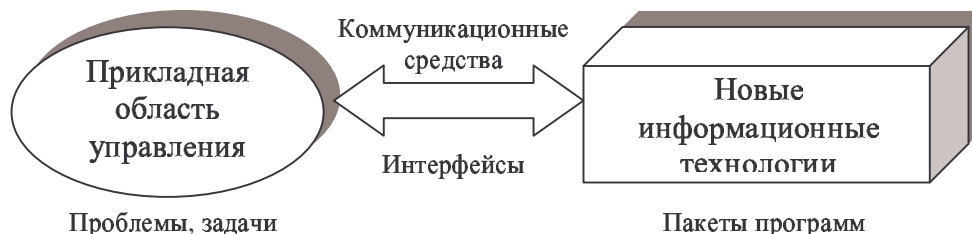
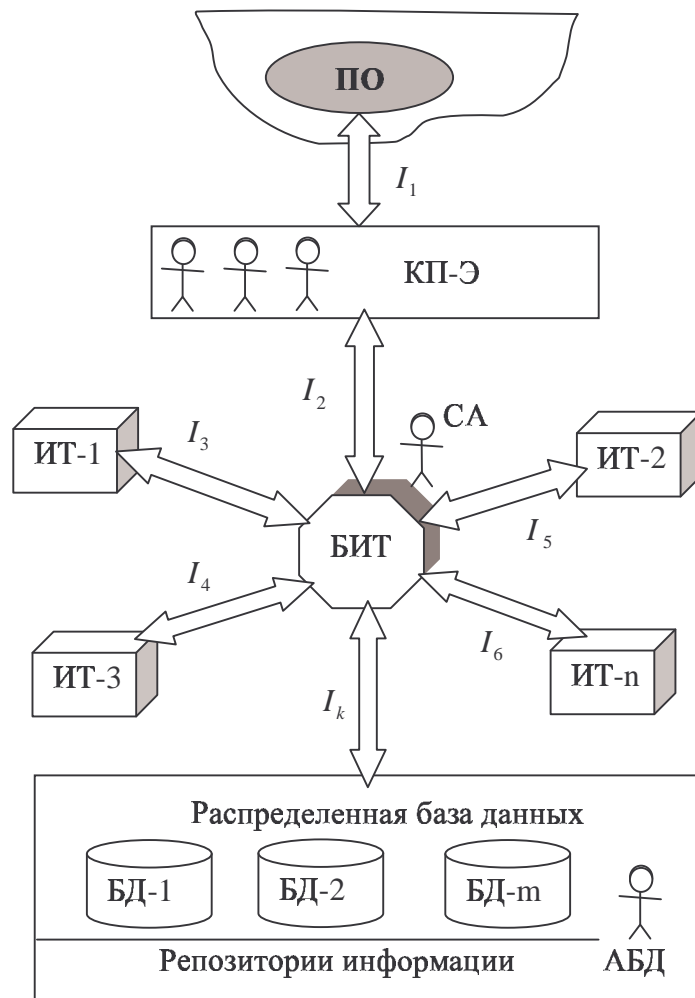


Рис.1. Общая схема ИИСУ

На рис.2. представлена развернутая обобщенная схема ИИСУ. Проблемная область (ПО) управляется конечными пользователями-экспертами (КП-Э), которые являются руководителями или специалистами данной прикладной области. I_1 - язык общения, интерфейс между ПО и КП-Э.

Рис.2.



Базовая информационная технология (БИТ) представляет собой основную платформу ИИСУ, это операционная система (напр. Windows-2000 или NT) и язык высокого уровня, обладающий объектно-ориентированными методами и средствами (напр. Borland C++ Builder, Visual C++ или др.).

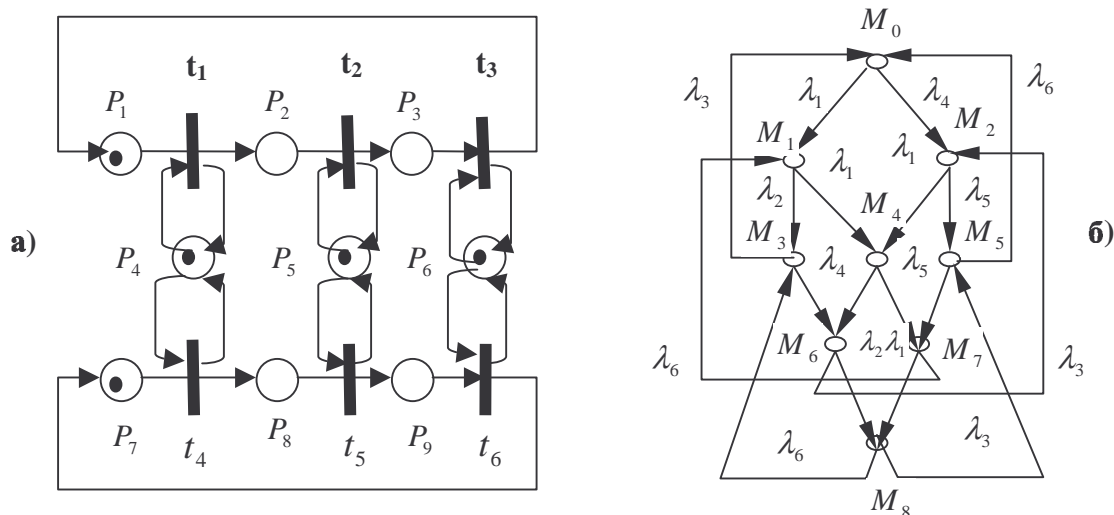
Информационные технологии (ИТ- i , $i=1, n$) представляются в широком спектре. При проектировании конкретной ИИСУ на этапе объектно-ориентированного анализа уточняется состав необходимых программных пакетов (информационных технологий) и коммуникационных программ (интерфейсов) между этими пакетами. Например, в качестве ИТ можно подобрать Web-технологии (HTML source coding, Macromedia Flash, Dreamweaver, Microsoft FrontPage и др.), Java Programming, JavaScript, MS-Office (Excel, Word, Access, Query и др.), DBase, Paradox (локальные БД), Oracle, SyBase, InterBase (распределенные БД) и т.д. [3,4,5,6].

С целью надежного и бесперебойного функционирования ИИСУ за ней следят системный администратор (СА) и администратор распределенной базы данных (АБД). Группа разработчиков ИИСУ обеспечивает систему стандартными или новыми разработанными интерфейсными драйверами, которые способны конвертировать данные из одного формата в другой, и этим самым обеспечить импорт-экспорт файлов между разными информационными технологиями. Например, драйвер-ODBC, который представляет собой стандартный DLL-файл, обеспечивает доступ из Excel (через MS Query) к внешним базам данных формата Dbase, SQL-Server, Access и др.

Если нет соответствующего интерфейсного драйвера между ИТ, то следует его разработать. Желательно его создать с помощью базового языка программирования, напр., на Borland C++Builder.

Одним из самых важнейших моментов при построении ИИСУ является вопрос проектирования распределенной базы данных и выбор подходящей СУБД. Сегодня актуальным является создание БД, типа клиент-сервер и многоярусных БД. В подобных распределенных системах используются технологии COM, CORBA или WEB.

В многопользовательских мультипроцессорных ИИСУ всегда возникает проблема определения оптимальных характеристик распределенной клиент-серверной сети для эффективного управления вычислительными ресурсами. С этой целью используются методы массового обслуживания (для определения количественных характеристик сети в стационарном режиме) и стохастических сетей Петри (для исследования динамических процессов событий сети) [7]. Рассматривается клиент-серверная сеть (с двумя серверами и тремя клиентами). На рис.3-а. представлен пример такой стохастической сети Петри, в которую кроме позиций и переходов добавлены и временные моменты открытия переходов (λ - ожидания). Серверы $S_1(P_1, P_2, P_3)$ и $S_2(P_7, P_8, P_9)$ готовы к обслуживанию клиентов, если в них имеются маркеры. Допустим, что в позициях клиентов $C(P_4, P_5, P_6)$ всегда имеются запросы на обслуживание (присутствие маркеров). $t_j, j = \overline{1,6}$ иллюстрируют времена открытия переходов (время обслуживания). Среднее время открытия перехода соответствует $1/\lambda$ (λ - интенсивность открытия переходов). На рис.3-б задан граф достижимости сети Петри. Количественный анализ подобной стохастической сети Петри можно осуществить с



помощью марковских процессов [8].

Рис.3.

Для определения вероятностей переходов системы из одного состояния в другое, следует построить систему уравнений Колмогорова и разрешить ее методом Гауса при заданных конкретных значениях интенсивностей λ .

Теперь рассмотрим пример разработки интерфейса между программными пакетами Ms-Excel и распределенной СУБД InterBase. Последняя система поддерживается фирмой Borland и она приемлема для C++ Builder [9]. Ставится задача конвертирования файлов (импорт/экспорт) из формата Excel в формат InterBase. В этом случае драйвер ODBC не пригоден. Следует автоматизировать процесс заполнения распределенной БД из отдельных электронных таблиц с целью дальнейшей централизации единой репозитории данных. На рис.4 представлен граф сети Петри, отображающий последовательность выполняемых процедур при функционировании интерфейсного драйвера[10]. На рис.5 и 6 представлены, соответственно фрагмент C++ программы файла импорт/экспорта, реализованной на базе Borland C++Builder и интерфейсное окно пользователя.

Следующим важным интерфейсом ИИСУ является язык структурных запросов-SQL пользователей (КП-Э, СА, АБД) системы, который принят почти во всех современных СУБД

(Oracle, SQL-Server, InterBase и т.д.). На рис.7. представлена диаграмма классов для ИИСУ, построенной с помощью UML-технологии, инструментом Paradigm_Plus [11].

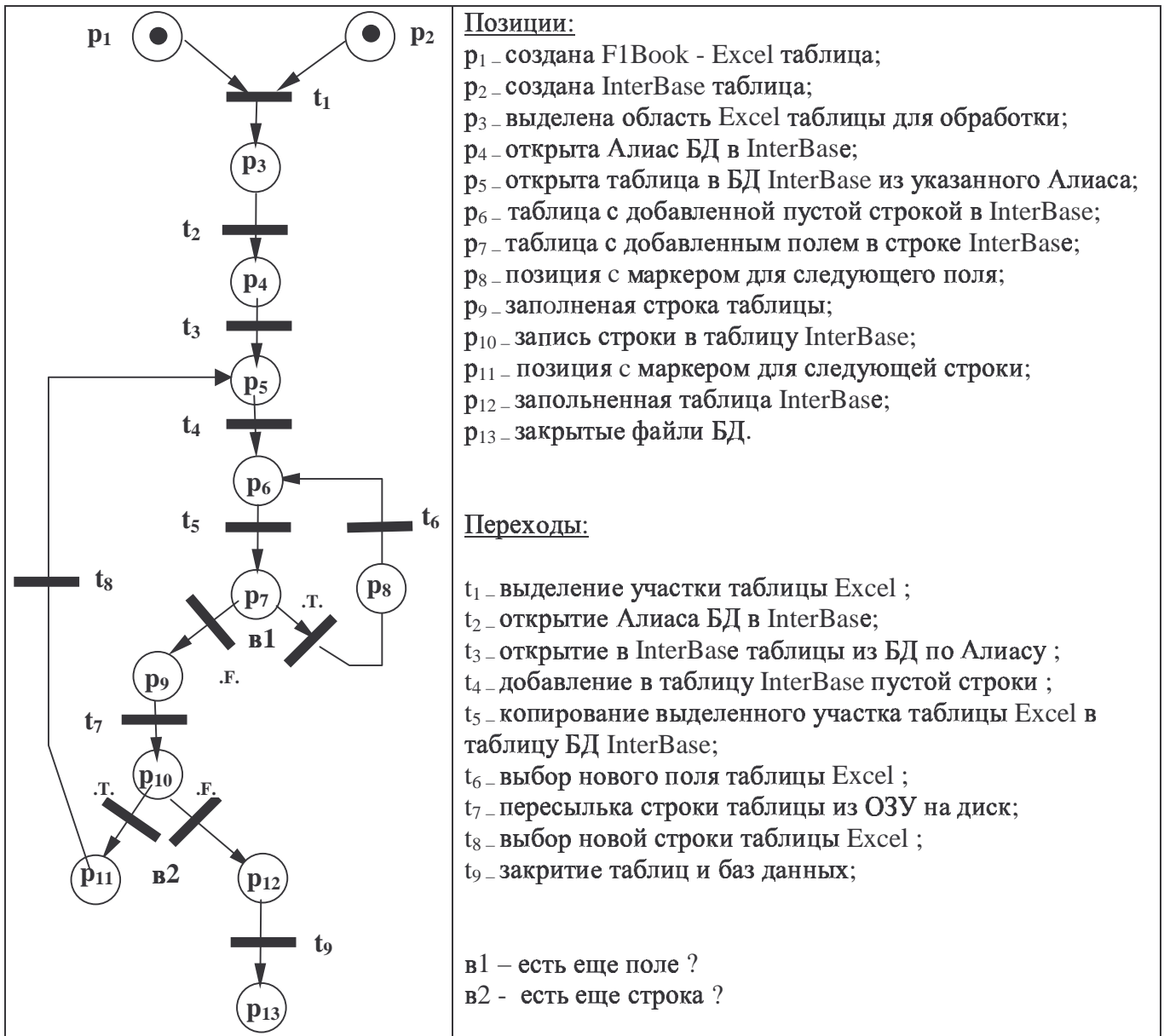


Рис.4. Граф сети Петри для кнопки ButtonClick: импорт данных из таблицы MsExcel – в базу данных InterBase

```
//----- Unit1.cpp ----- текст программы для рис 5 -----
int i,j, Ob_Num, Dan_Num, Rec_Num;
void __fastcall TForm1::FormCreate(TObject *Sender)
{ Session->GetAliasNames(cbAlias->Items); }
void __fastcall TForm1::cbAliasChange(TObject *Sender)
{ Session->GetTableNames(cbAlias->Text, "", true, false, cbTable->Items);
cbTable->ItemIndex = 0;
if (cbTable->Text == "")
{ ShowMessage("Table not selected"); return; }
Table1->Active = false;
Table1->DatabaseName = cbAlias->Text;
Table1->TableName = cbTable->Text;
Table1->Active = true;
if(Table1->Active) Table1->GetFieldNames(cbField->Items);
```

```

}
//----- Button of Import -----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
  AnsiString Info;
  if(Table1->Active)
  {
    int t=Table1->FieldCount;
    for(i=3; i<Rec_Num+3; i++) // str_num
    {
      Table1->Append();
      for(j = 1; j < t; j++) // col_num
      {
        Label3->Caption=t;
        Table1->FieldValues["OBJNUM"] = Ob_Num;
        Info = Table1->Fields->Fields[j]->FieldName;
        Label1->Caption=j;
        Excel->SetSelection(i,j,i,j); // choice of Range
        if(Excel->Text != 0)
        {
          Label2->Caption=Excel->Text;
          Table1->FieldValues[Info] = Excel->Text;
        }
      } // for j
      Table1->Post();
    } // for i
  } // for if()
}

```

Рис.5. Программный фрагмент .cpp файла интерфейсного драйвера

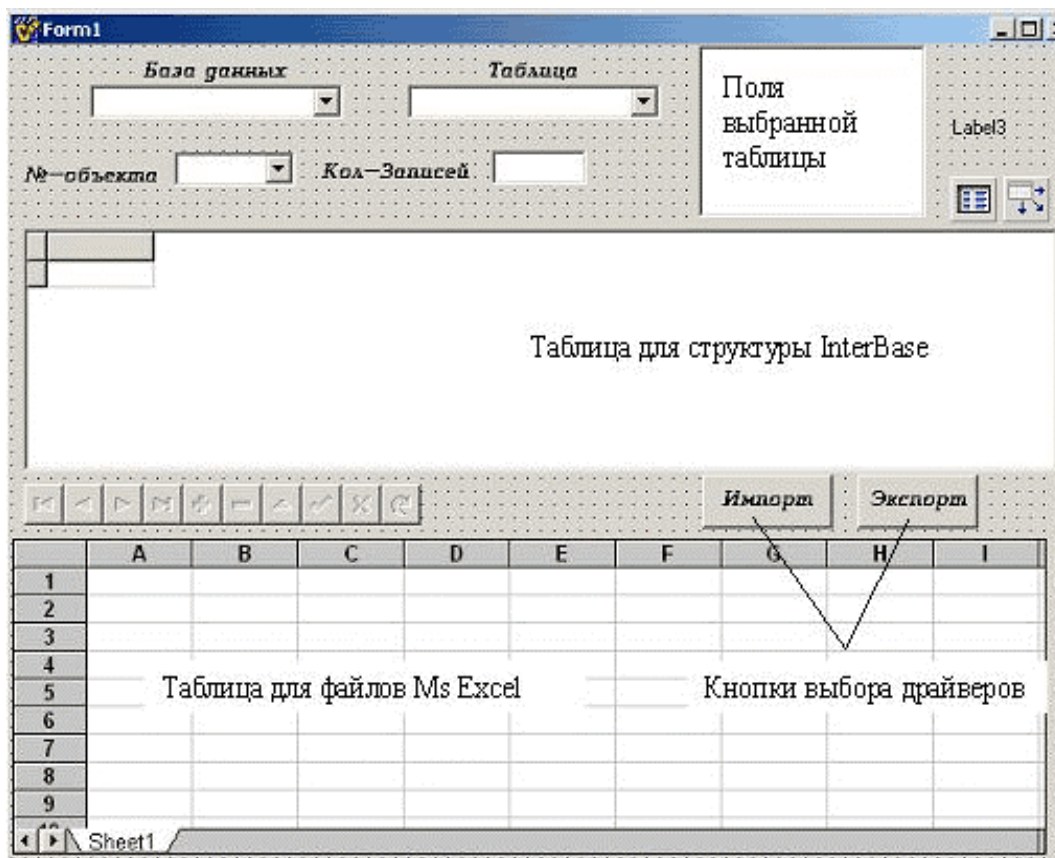


Рис. 6. Интерфейс для преобразования форматов таблиц

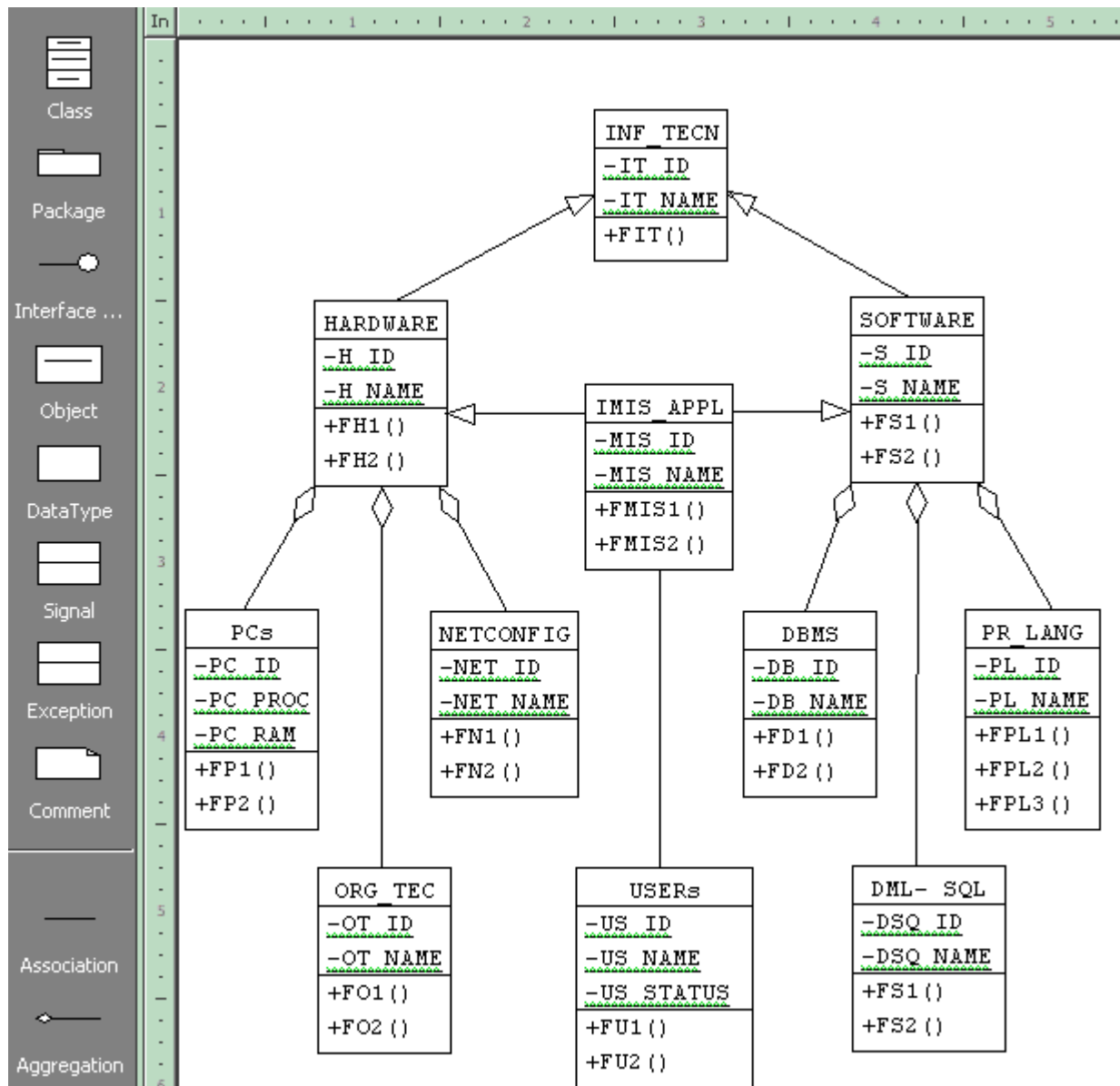


Рис.7. Фрагмент диаграммы классов в рабочей среде Paradigm_Plus

При проектировании конкретной ИИСУ конкретизируются и детализируются все атрибуты и функций указанных классов. Построены и другие диаграммы, характерные для UML: UseCase, Interaction, State Transition, Activity, Component и т.д. [12].

Литература

1. Боттэ К., Сургуладзе Г., Кашибадзе М. Наследственность в программировании информационных систем управления: от систем баз данных до UML-технологии. Тр.ГТУ, 2001, №4(437), с.55-62.
2. G. Booch, I. Jacobson, J. Rumbaugh. Unified Modeling Language. User Guide, Addison Wesley Longman, 1999.
3. Матросов А.С., Сергеев А.О., Чаунин М.П. HTML 4.0: Новый уровень создания HTML-документов. Санкт-Петербург, 1999.
4. Morgan M. Java-2 for Professional Developers. Пер. с англ., Москва, 2000.
5. Грин Дж. и др. Oracle 8/8i Server. Энциклопедия пользователя. Пер. с англ. Киев, 2000.
6. InterBase 5. Programmer's Guide. Software Corp., CA-95066. www.interbase.com.
7. Болх Г., Сургуладзе Г., Петриашвили Л., Чихрадзе Б. Разработка программного обеспечения системы управления ресурсами мультипроцессорных систем на базе BorlandC++Builder. Тр.ГТУ, 2001, №4(437), с.47-53.
8. Bolch G., Greiner S., de Meer H., Trivedi K. Queueing Networks and Markov Chains: Modeling and Performance Evaluation With Computer Science Applications, JOHN WILEY & SONS, INC.1998.
9. Архангельский А.Я. Программирование в C++Builder. Москва, 2001.
10. Reisig W. Elements of Distributed Algorithms: Modeling and Analysis with Petri Nets. Springer, Berlin, 1998.
11. Paradigm Plus. Documentation and computer software program for user's. Computer Associates International (CA), 2001.
12. Сургуладзе Г., Долидзе Т., Туркия Е. Проектирование и реализация информационных систем управления маркетингом производственных фирм на базе UML-технологии. Тр.ГТУ, 2001, №4(437), с.193-200.