



Microsoft
GOLD CERTIFIED
Partner

Реализация RAD системы Clarion в среде .NET

Arcadia, Inc.
12 Podyezdnoy pereulok, St. Petersburg 190013 Russia
Tel: +7 (812) 764 84 56, 712 40 09, 572 78 11
Fax: +7 (812) 712 40 18
E-mail: info@arcadia.spb.ru
info@arcadia.spb.ru
<http://www.arcadia.spb.ru>
<http://www.offshore-software.ru>

План доклада

- ▶ Быстрая разработка приложений (RAD)
- ▶ Архитектура среды разработки Clarion
- ▶ История Clarion
- ▶ Особенности языка Clarion
- ▶ Архитектура компилятора Clarion.NET
- ▶ IDE Clarion.NET

Clarion (Кларион) - духовой инструмент, издающего звук предельно высокой частоты, и впервые появившегося в 14 веке использовавшаяся в средние века как сигнал во время военных действий

Clarion — это ориентированная на обработку данных среда быстрой разработки приложений, которая включает кодогенерацию и многократно используемые метаданные для быстрого создания приложений уровня управления предприятием.

- ▶ Среда быстрой разработки приложений баз данных
- ▶ Язык программирования четвертого поколения ориентированный на разработку баз данных
- ▶ Основан в 1986 г Брюсом Барингтоном, владельцем фирмы медицинского оборудования HBO & Company
- ▶ Проблемно-ориентированный и не зависимый от конкретной базы данных
- ▶ Драйвера для различных баз данных: SQL, Clipper, FoxPro, ASCII, TopSpeed, Oracle
- ▶ Включает в себя черты языка общего назначения

Быстрая разработка приложений (Rapid Application Development)

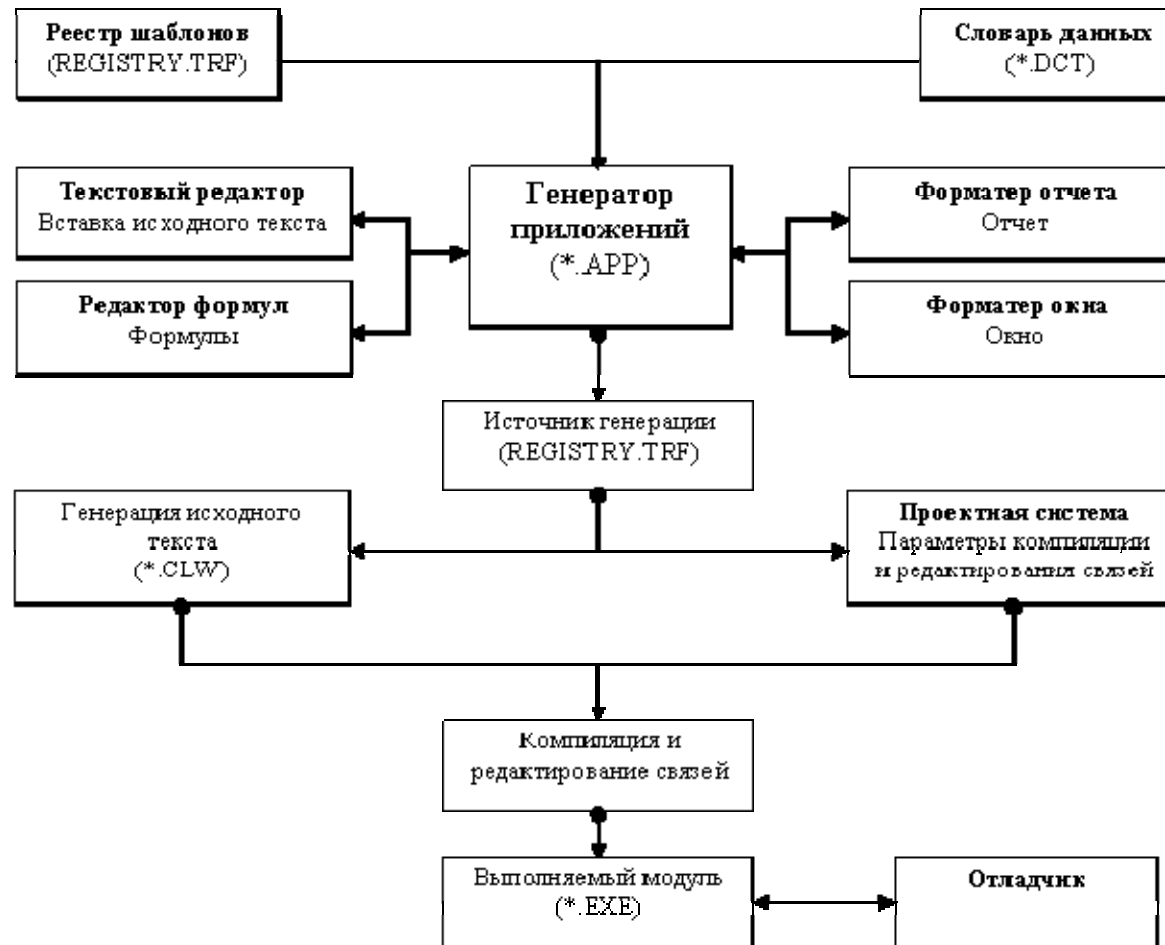


Методология RAD:

- ▶ Основатель Джеймс Мартин, IBM , 1980-е гг
- ▶ Технологический процесс для максимально быстрой разработки приложений
- ▶ Применение средств визуального программирования
- ▶ Итерационная (спиральная) модель разработки
- ▶ Максимальная автоматизация рутинных операций
- ▶ KISS – Keep it Simple, Stupid
- ▶ Обильное создание прототипов
- ▶ Тесное взаимодействие с заказчиком

Архитектура Clarion

Последовательность разработки приложения

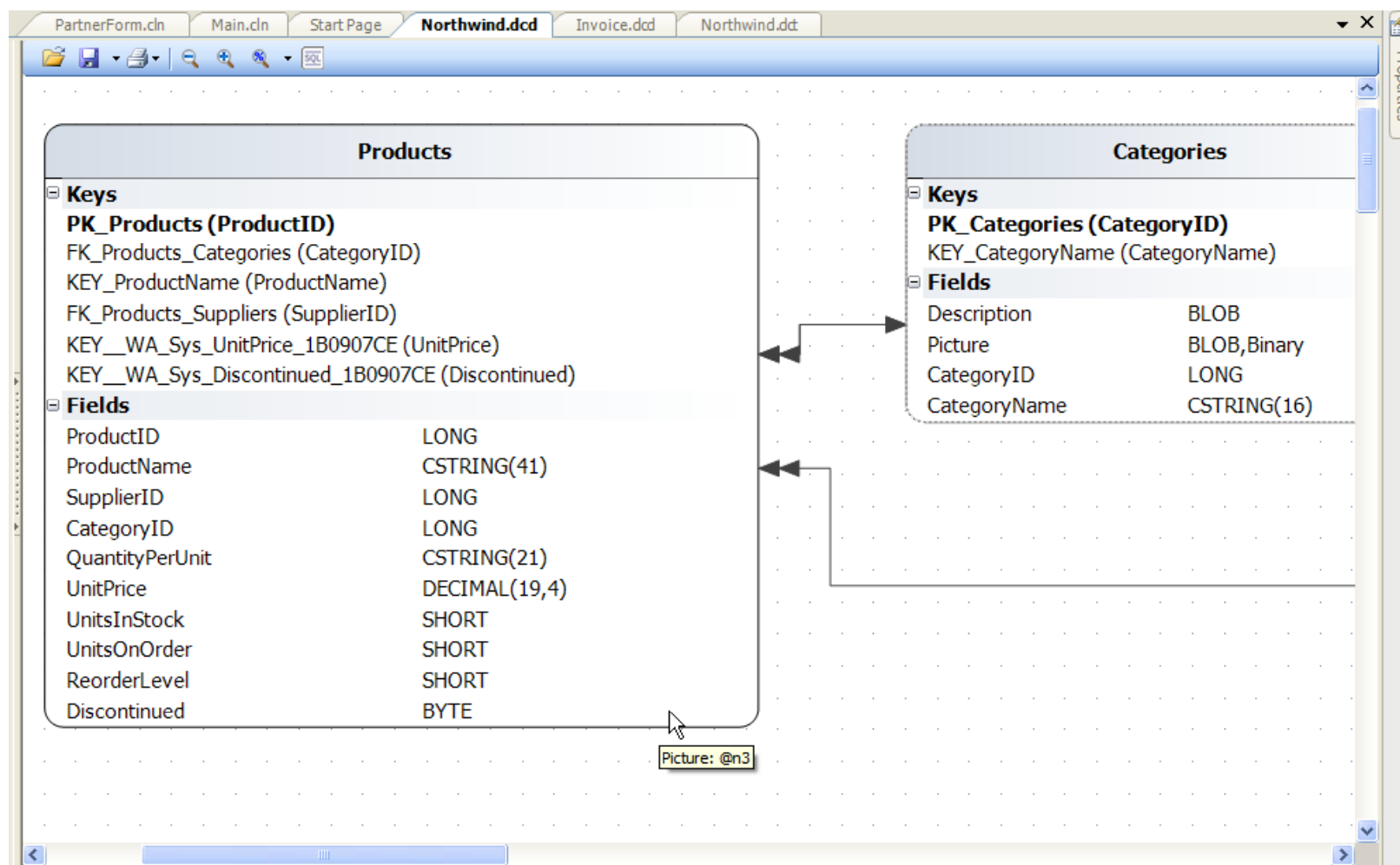


Архитектура Clarion



- ▶ Словарь данных (Data Dictionary)
- ▶ Генератор приложений (Application Generator)
- ▶ Редактор отчетов и окон
- ▶ Текстовый редактор
- ▶ Шаблоны (Templates)
- ▶ Язык Clarion

Словарь данных (Data Dictionary)



Словарь данных (Data Dictionary)

- ▶ Ядро системы Clarion
- ▶ Описывает таблицы баз данных
- ▶ Связи между таблицами
- ▶ Специальные атрибуты для представления данных на экране, бизнес правила и опции использования
- ▶ Используется как основа для генератора приложений

Генератор приложений (Application Generator)



- ▶ Генерирует приложение на основе метаданных
- ▶ Возможность модификации сгенеренного кода в точках вставки
- ▶ При генерации кода используются шаблоны
- ▶ Полное отсутствие кодирования для стандартных задач

Шаблоны (Templates)

- ▶ Гибко настраиваемые, с диалоговым интерфейсом, интерпретируемые скрипты для генерации кода
- ▶ Язык шаблонов
- ▶ Предлагают ответить на некоторые вопросы, на основании ответов на которые строится определенный набор модулей с исходным кодом.
- ▶ Обеспечивают многократное использование и централизованное обслуживание кода.

История Clarion



1982г	Clarion Software Corporation (Брюс Барингтон)
1986г	1.0 DOS
1987г	Jensen and Partners International (JPI) - Нильс Йенсен
1988г	2.0 Clarion Professional Developer (CPD)
1992	TopSpeed Corporation
1993г	3.0 Clarion Database Developer (CDD)
1994г	Clarion for Windows (CW1.0)
1997г	CW2.0
1998г	Clarion 4 – ABC библиотека
2000г	Clarion 5
2000г	SoftVelocity , Inc, Robert Zaunere (Роберт Заунер)
2002г	Clarion 6
2005 г	Arcadia Inc. , начало разработки Clarion.NET и IDE
2007г	Clarion 7(beta) – SharpDevelop IDE

Особенности языка Clarion



- ▶ За основу Clarion взяты языки:
COBOL, ADA, BASIC
- ▶ Работа с файлами баз данных:
FILE, KEY, VIEW, INDEX, BLOB
- ▶ Визуальное представление:
APPLICATION, WINDOW, REPORT, UI Controls
- ▶ Типы данных: CLASS, QUEUE, GROUP
- ▶ Обработчик событий: ACCEPT LOOP, события
- ▶ Препроцессор: EQUATE, OMIT
- ▶ Неявное преобразование типов

MDI application

```
1  PROGRAM
2  MAP
3  Main PROCEDURE
4  Hello PROCEDURE
5  EventString FUNCTION (LONG PassedEvent), STRING
6      END
7  CODE
8  Main PROCEDURE
9  AppFrame APPLICATION (.Hello Windows.), AT (, , 280, 200), SYSTEM, RESIZE, MAX
10     MENUBAR
11         MENU (.&File.), USE (?File)
12             ITEM (.&Browse Phones.), USE (?FileBrowsePhones)
13             ITEM, SEPARATOR
14             ITEM (.E&xit.), USE (?FileExit), STD (STD:Close)
15         END
16         ITEM (.&About.), USE (?About)
17     END
18 END
19 CODE
20     OPEN (AppFrame)
21     ACCEPT
22     CASE ACCEPTED ()
23     OF ?About
24         Hello
25     END
26 END
```

FILE, WINDOW

```
56
57 Teachers FILE, DRIVER('TOPSPEED'), PRE(TEA), NAME('TPS\TEACHERS.TPS'), CREATE, BINDABLE, THREAD
58 KeyTeacherNumber KEY(TEA:Number), NOCASE, OPT, PRIMARY
59 KeyLastName KEY(TEA:LastName), DUP, NOCASE
60 KeyDepartment KEY(TEA:Department), DUP, NOCASE, OPT
61 Record RECORD, PRE()
62 Number SIGNED
63 FirstName CLASTRING(20)
64 LastName CLASTRING(20)
65 Address CLASTRING(20)
66 City CLASTRING(20)
67 State CLASTRING(2)
68 Zip SIGNED
69 Telephone CLASTRING(12)
70 Department SIGNED
71 END
72 END
73
74 MDIChild2 WINDOW('Dialog Window'), MDI, SYSTEM, MAX, STATUS
75 MENUBAR, NOMERGE
76 MENU('File'), USE(?FileMenu)
77 ITEM('Close'), USE(?CloseFile)
78 END
79 MENU('Edit'), USE(?EditMenu)
80 ITEM('Undo'), USE(?UndoText), KEY(CtrlZ), STD(STD:Undo)
81 ITEM('Cu&t'), USE(?CutText), KEY(CtrlX), STD(STD:Cut)
82 ITEM('Copy'), USE(?CopyText), KEY(CtrlC), STD(STD:Copy)
83 ITEM('Paste'), USE(?PasteText), KEY(CtrlV), STD(STD:Paste)
84 END
85 END
86 TEXT, HVSCROLL, USE(Pre:Field), MSG('Enter some text here')
87 BUTTON('&OK'), USE(?Exit), DEFAULT
88 END
```


ACCPT LOOP, QUEUE

```
73
74 OPEN (Window)
75 ACCEPT !Event handler
76 CASE FIELD()
77 OF 0 !Handle Field-independent events
78 CASE EVENT()
79 OF EVENT:Move
80 CYCLE !Do not allow user to move the window
81 OF EVENT:Suspend
82 CASE FOCUS()
83 OF ?Field1
84 !Save some stuff
85 END
86 OF EVENT:Resume
87 !Restore the stuff
88 END
89 OF ?Field1 !Handle events for Field1
90 CASE EVENT()
91 OF EVENT:Selected
92 ! pre-edit code for field1
93 OF EVENT:Accepted
94 ! completion code for field1
95 END
96 END
97
98 NameQue QUEUE, PRE (Nam) !Declare a queue
99 Name STRING (20)
100 Zip DECIMAL (5,0), NAME ('SortField')
101 END !End queue structure
102
103 NameQue2 QUEUE (NameQue), PRE (Nam2) !Queue that inherits Name and Zip
104 Phone STRING (10) ! and adds a Phone field
105 END
106
107 NameQue3 NameQue2 !Declare a second QUEUE with
108 !exactly the same structure as NameQue2
```

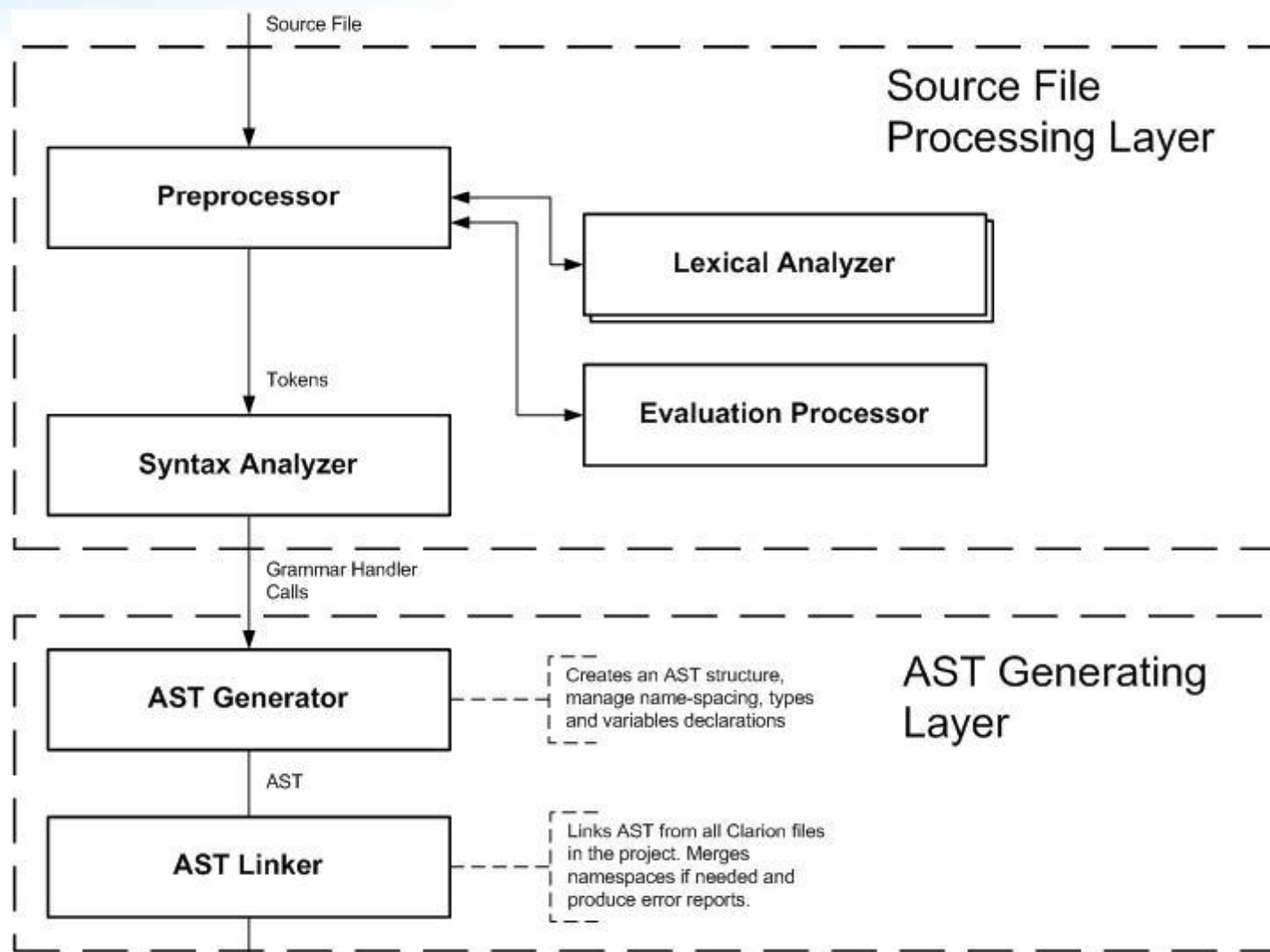
Язык Clarion.NET

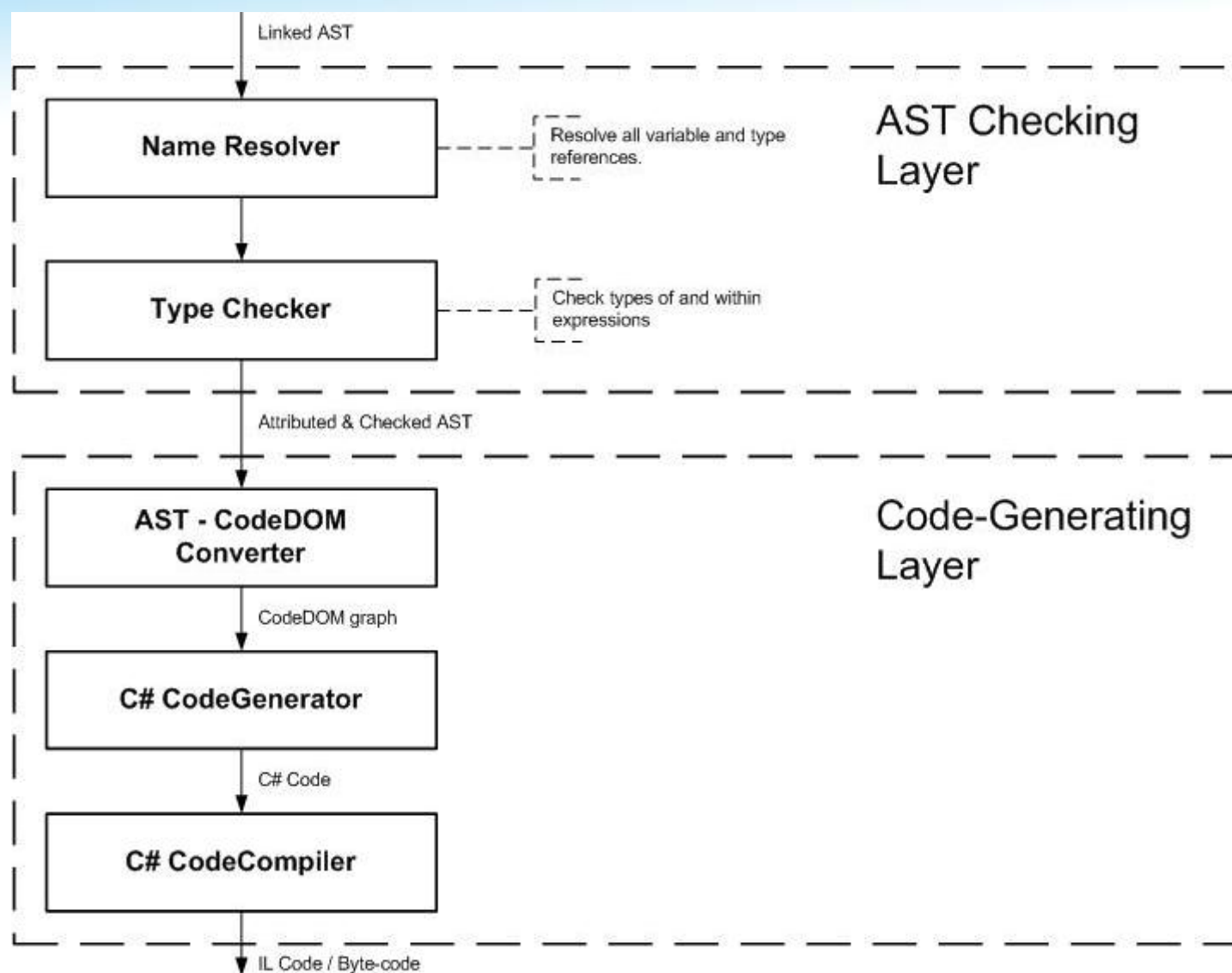
Язык Clarion был несколько модифицирован, в соответствии с требованиями, предъявляемыми средой .Net

- ▶ Директива NAMESPACE (аналог **namespace** in C#)
- ▶ Директива USING (аналог **using** in C#)
- ▶ Поддержка property, indexer, event, delegate
- ▶ Конструкция FOREACH (аналог **foreach** in C#)
- ▶ Поддержка исключений (TRY ... CATCH)
- ▶ Поддержка .Net's custom attributes

Структура компилятора

- ▶ Лексический анализатор. Разбивает исходный код на лексемы.
- ▶ Препроцессор
- ▶ Синтаксический анализатор. Производит синтаксический разбор и строит абстрактное синтаксическое дерево программы (AST)
- ▶ Обработчик абстрактного синтаксического дерева (AST). Обеспечивает разрешение ссылок на объекты программы (Name Resolving), проверку корректности типов (Type Checking) etc.
- ▶ Генератор кода. Генерирует окончательный код из обработанного синтаксического дерева.



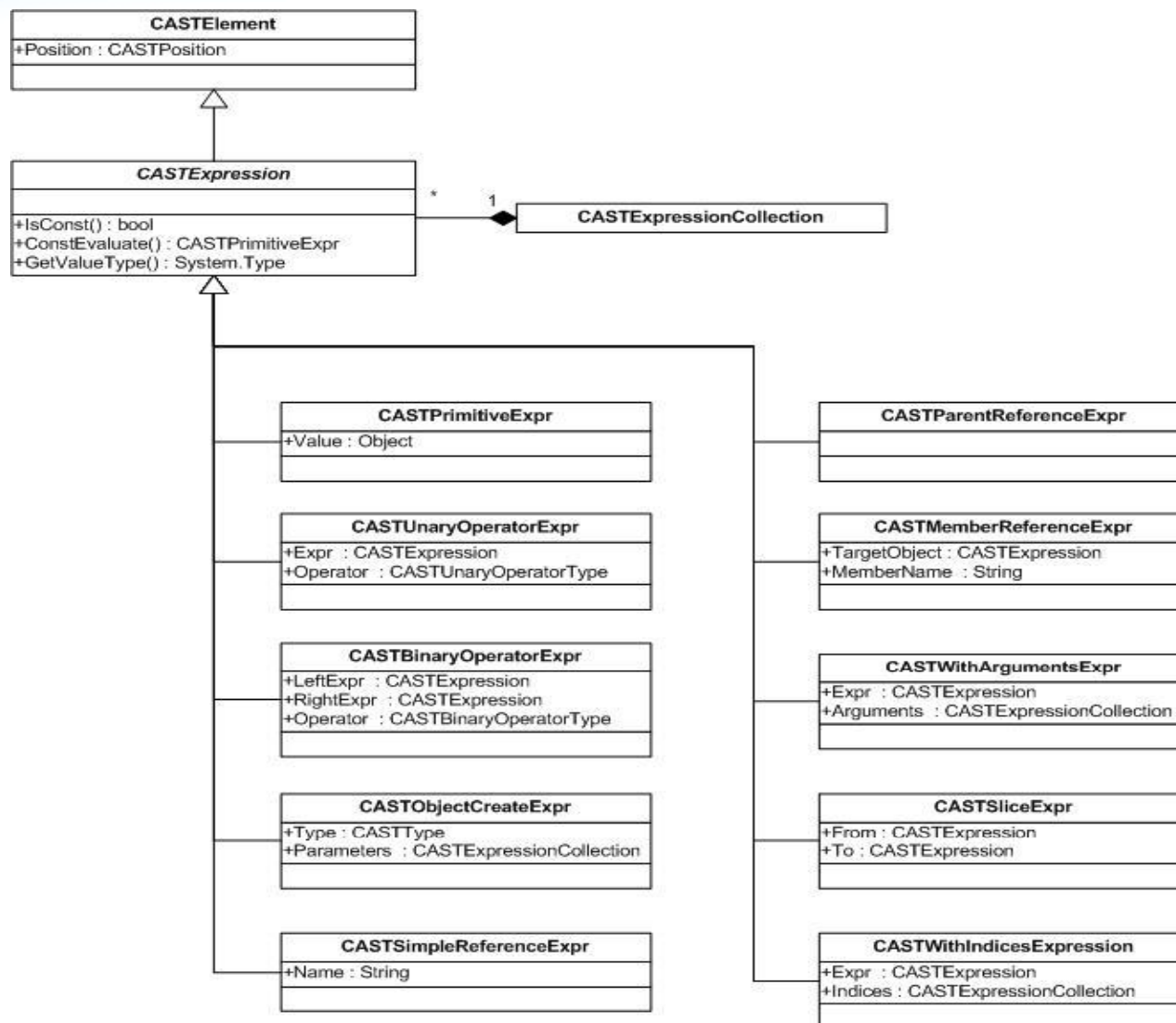


Синтаксический анализатор



- ▶ Лексический анализатор
- ▶ Препроцессор: ASSERT, EQUATE, OMIT, INCLUDE, Pragmas
- ▶ LALR(1) (Look Ahead Left to Right) грамматика
- ▶ Генерация синтаксического дерева AST (abstract syntax tree)

AST



Разрешение ссылок (Name Resolving)

- ▶ Поиск объектов (переменная, процедура,...) по имени в текущем контексте
- ▶ Поиск объектов во внешних сборках
- ▶ Проверка семантической правильности AST дерева

- ▶ Классы, интерфейсы, и архитектура для представления структуры исходной программы, независимую от языка программирования
- ▶ Провайдеры для генерации кода на разных языках
- ▶ Clarion CodeDOM provider
- ▶ Snippets

Генерация кода (Code Generation)

- ▶ Создание Code-DOM дерева на основе AST дерева
- ▶ Генерация кода на языке C# с помощью C# CodeDOMprovider
- ▶ Возможность генерации кода в IL

Runtime libraries

- ▶ Встроенные функции
- ▶ Стандартные классы REPORT, QUEUE, VIEW
- ▶ WINDOW структура и ACCEPT LOOP (не реализованы)
- ▶ Обертка для драйверов баз данных

- ▶ SharpDevelop 2.0 - свободно распространяемое IDE с открытым кодом
- ▶ Поддержка языков C#, VB.NET, Boo
- ▶ Позволяет расширять функциональность с помощью плагинов (plug-ins)
- ▶ Clarion 7 и Clarion .NET plug-ins
- ▶ Исправления в ядре SharpDevelop
- ▶ Отладчик

Интеграция IDE и компилятора



- ▶ Использует Clarion.NET компилятор для парсинга исходного кода и получения AST дерева
- ▶ На каждый файл проекта свое AST дерево
- ▶ Редактируемый файл разбирается каждые 2 сек
- ▶ IDE parser

Интеграция IDE и компилятора



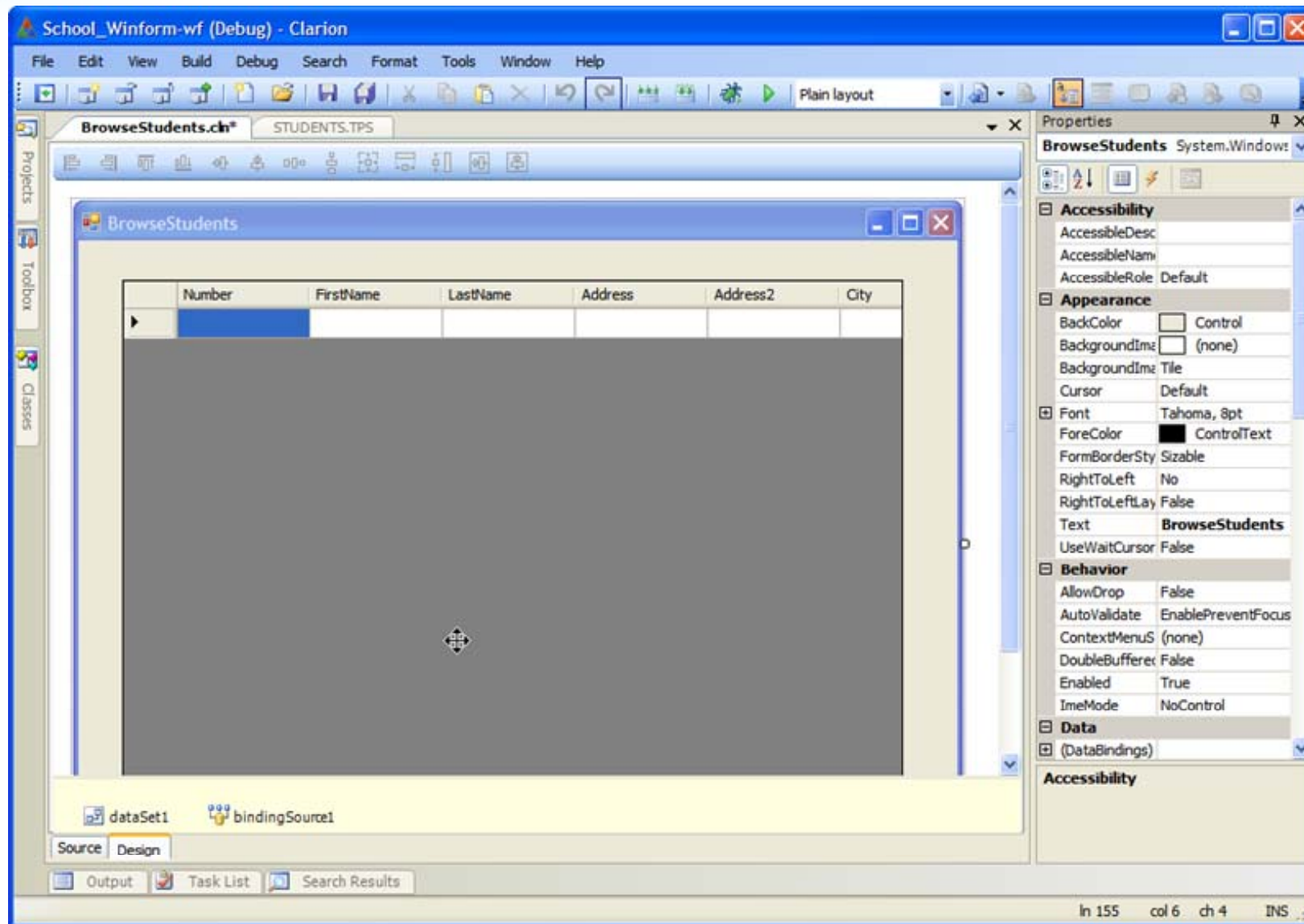
IDE features

- ▶ Code Completion
- ▶ Folding
- ▶ Class browser
- ▶ Quick class browser
- ▶ Code navigation
- ▶ Refactoring
- ▶ Tool tips
- ▶ Method insight

Designers

- ▶ Form designer
- ▶ Web form designer
- ▶ WINDOW/REPORT designer
- ▶ Compact Framework form designer

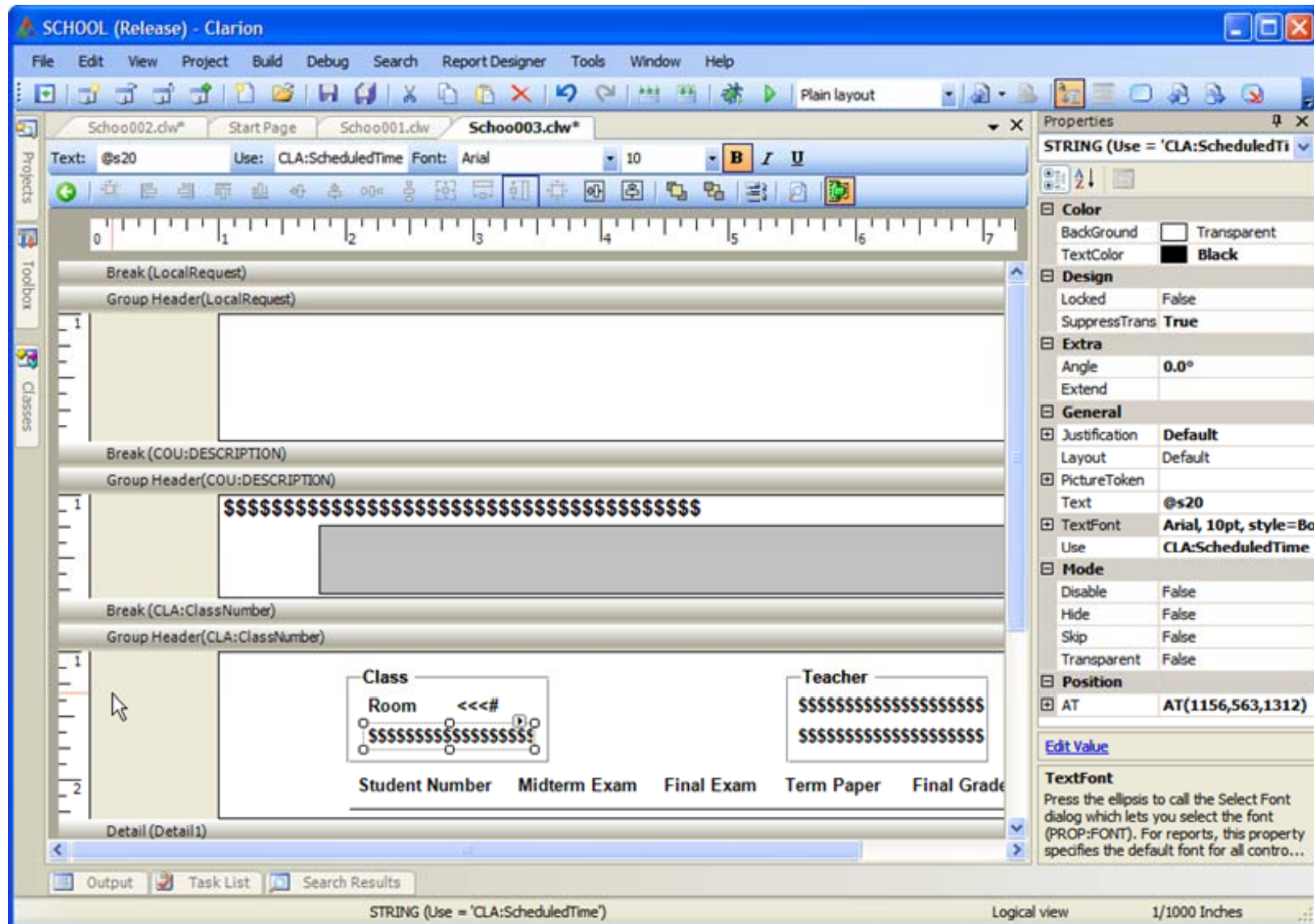
Form designer



Form designer

- ▶ Предназначенный для создания оконных приложений с минимальными затратами на написание кода.
- ▶ Поддерживает полный набор стандартных .NET элементов управления
- ▶ Автоматически генерирует исходный код на Clarion .NET
- ▶ UI designer -> CodeDOM -> Clarion.NET
- ▶ Clarion.NET -> AST -> CodeDOM -> UI designer

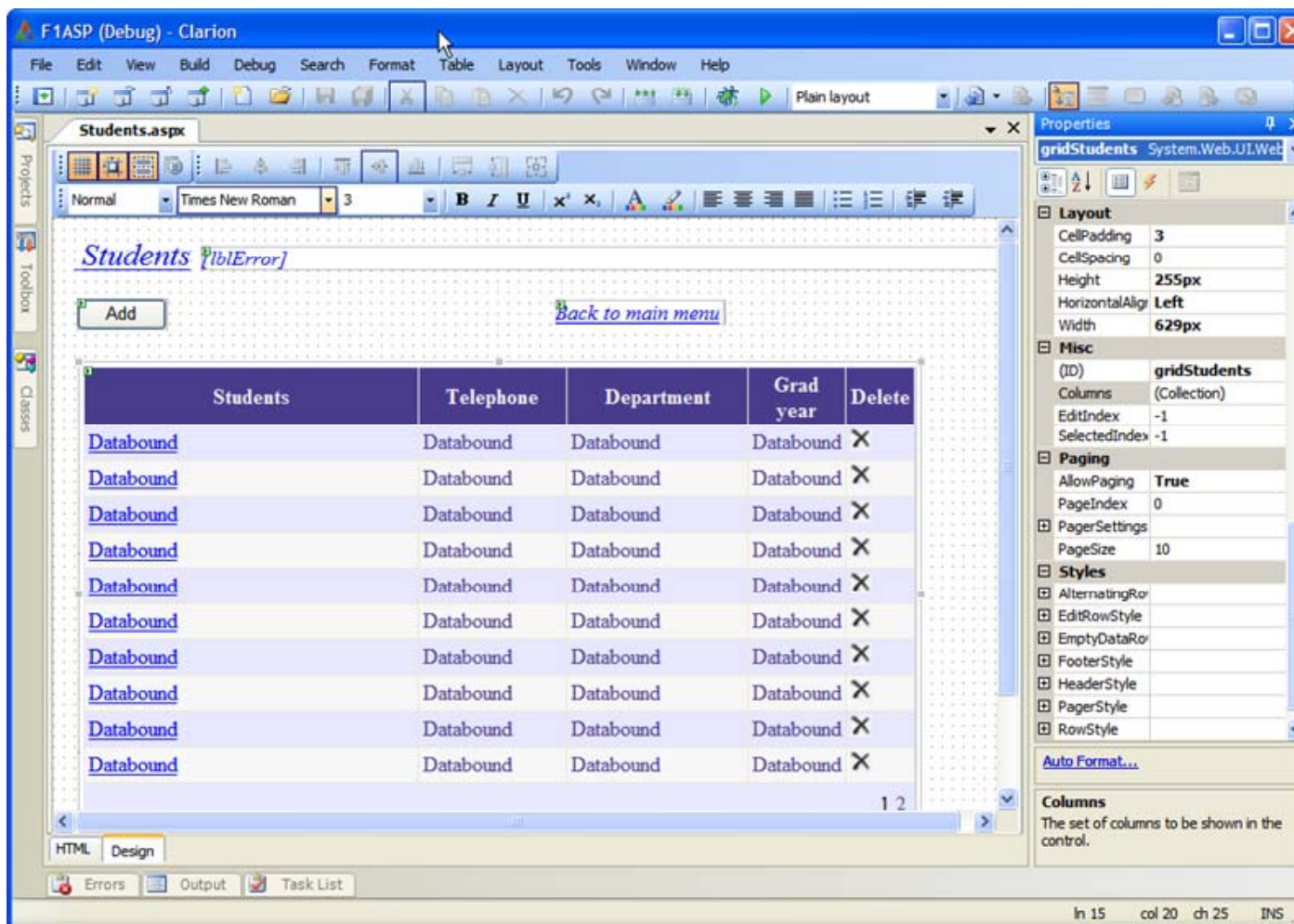
Report designer



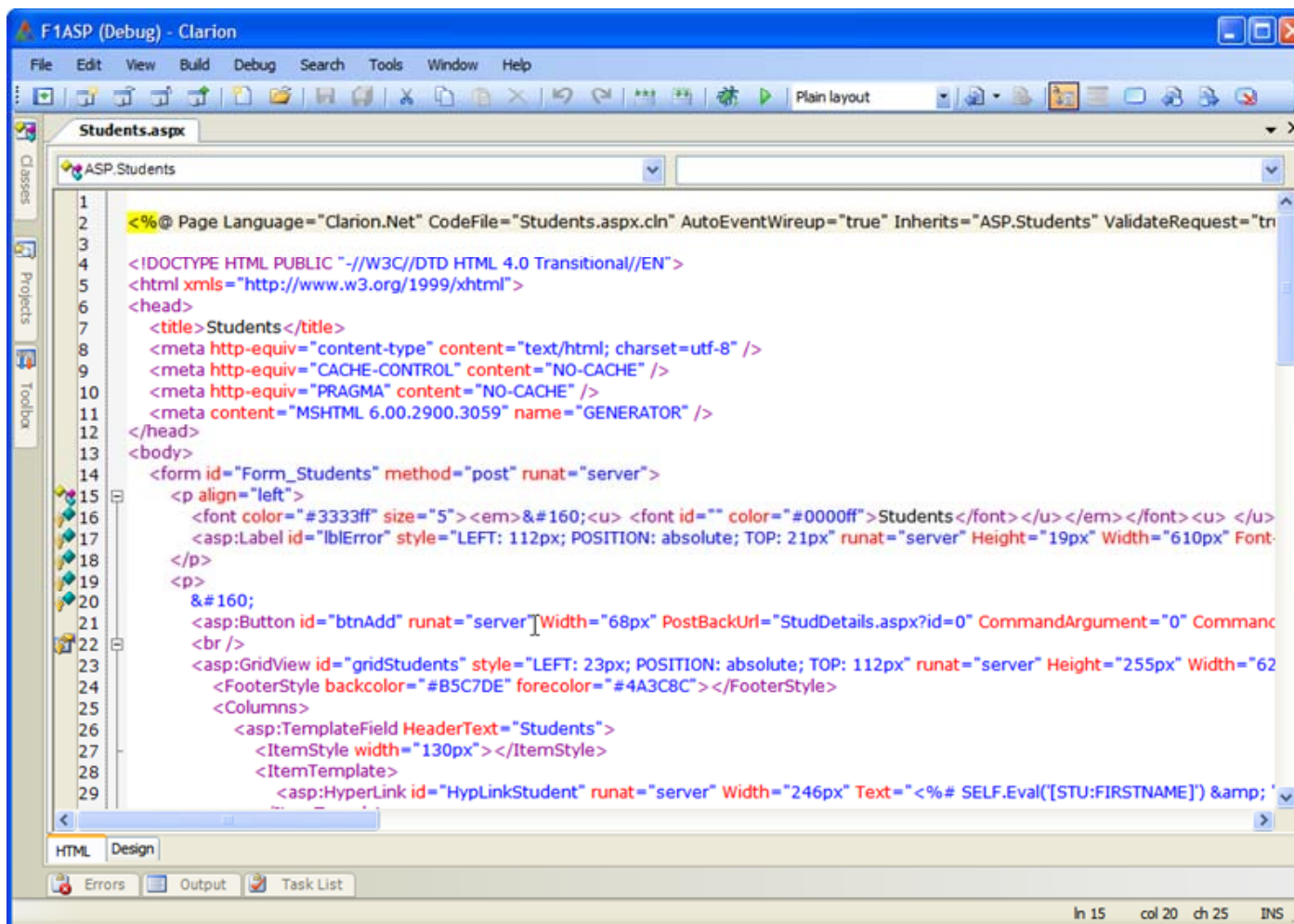
Report designer

- ▶ Визуальное создания и редактирования заготовок отчетов
- ▶ Автоматически генерирует исходный код в виде структуры REPORT
- ▶ UI designer -> REPORT
- ▶ REPORT -> AST -> UI designer

Поддержка ASP.NET



Поддержка ASP.NET

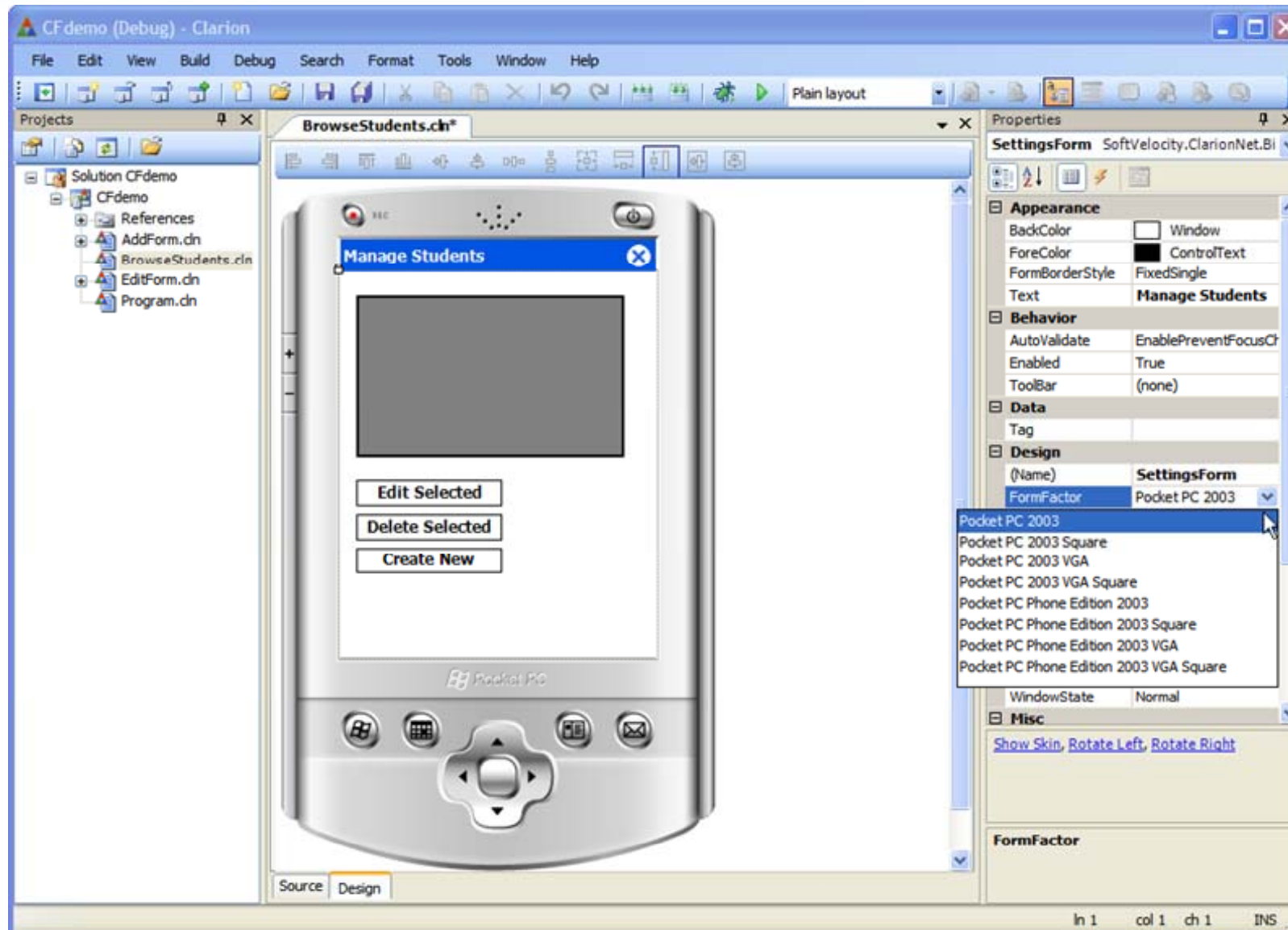


```
1  <%@ Page Language="Clarion.Net" CodeFile="Students.aspx.cln" AutoEventWireup="true" Inherits="ASP.Students" ValidateRequest="true" %>
2
3
4  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
5  <html xmlns="http://www.w3.org/1999/xhtml">
6  <head>
7      <title>Students</title>
8      <meta http-equiv="content-type" content="text/html; charset=utf-8" />
9      <meta http-equiv="CACHE-CONTROL" content="NO-CACHE" />
10     <meta http-equiv="PRAGMA" content="NO-CACHE" />
11     <meta content="MSHTML 6.00.2900.3059" name="GENERATOR" />
12 </head>
13 <body>
14     <form id="Form_Students" method="post" runat="server">
15         <p align="left">
16             <font color="#3333ff" size="5"><em>&#160;</em><u> <font id="" color="#0000ff">Students</font></u></em></font><u> </u></u></p>
17             <asp:Label id="lblError" style="LEFT: 112px; POSITION: absolute; TOP: 21px" runat="server" Height="19px" Width="610px" Font-
18             </p>
19             <p>
20                 &#160;
21                 <asp:Button id="btnAdd" runat="server" Width="68px"PostBackUrl="StudDetails.aspx?id=0" CommandArgument="0" Command
22                 <br />
23                 <asp:GridView id="gridStudents" style="LEFT: 23px; POSITION: absolute; TOP: 112px" runat="server" Height="255px" Width="62
24                 <FooterStyle bgcolor="#B5C7DE" forecolor="#4A3C8C"></FooterStyle>
25                 <Columns>
26                     <asp:TemplateField HeaderText="Students">
27                         <ItemStyle width="130px"></ItemStyle>
28                         <ItemTemplate>
29                             <asp:HyperLink id="HypLinkStudent" runat="server" Width="246px" Text="<%# SELF.Eval("[STU:FIRSTNAME]") &amp; "
```

Поддержка ASP.NET

- ▶ Визуальный дизайнер форм для веб-страниц (.aspx)
- ▶ Поддерживает полный набор стандартных ASP.NET элементов управления
- ▶ Генерация ASPX страниц с помощью Internet Explorer API
- ▶ Реализация частичных классов в Clarion.NET
- ▶ При запуске приложения ASP.NET runtime создает код на языке Clarion.NET с помощью Clarion CodeDOM провайдера
- ▶ Отладчик присоединяется к процессу wp_aspnet.exe

Compact Framework



Compact Framework

- ▶ Редактор форм для Compact Framework(CF)
- ▶ Поддержка проектов для (CF) в IDE
- ▶ Возможность использования стандартных библиотек и контролов для CF
- ▶ Не поддерживается отладка и запуск приложения на эмуляторе (MS Device Emulator)

Перспективы развития

- ▶ generics в компиляторе
- ▶ Переход на .NET framework 3.0
- ▶ Парсер для IDE
- ▶ Отладчик для Compact Framework
- ▶ Интеграция с VS2007 Shell
- ▶ AWT
- ▶ Интеграция IDE с генератором приложений

Contact Information



ARCADIA, INC.

St. Petersburg, Russia

Tel.: +7 812 764 84 56

Fax: +7 812 712 40 18

E-mail: info@offshore-software.ru

www.offshore-software.ru



ARDIN SOFTWARE OY

Vantaa, Finland

Tel.: + 358 40 820 60 57

Fax: + 358 9 872 59 68

E-mail: info@ardinsoftware.fi

www.ardinsoftware.fi



PLANET-SOFTWARE

Gainesville, Florida, US

Tel.: +1 352-222-2092

E-mail:

planetsoftware@compuserve.com

www.planet-software.com