

Capítulo 1

Introdução

(R)evolução

“(R)evolução”. Amigos, esta é a palavra mais falada pelos desenvolvedores Delphi quando fazem o primeiro contato com a versão 7 do produto. Uns dizem **evolução**, outros **revolução**. Particularmente acho que o Delphi amadureceu de uma forma extraordinária, e que a equipe de negócios do produto conseguiu quebrar algumas barreiras importantes, como a incorporação de um excelente gerador de relatórios, ou melhor, um poderoso conjunto de componentes conhecido por grande parte dos desenvolvedores Delphi: Rave Report.

Como o foco principal deste livro é o desenvolvimento de aplicações e-Business, a maior novidade para os Web-Developers é o excelente pacote da empresa Atozed. Já ouviram falar em IntraWeb? Agora sim temos uma ferramenta RAD de respeito para o desenvolvimento de aplicações Web. Veja que não quero desprezar a tecnologia existente desde a versão 3 do produto, nem a nova tecnologia WebSnap, introduzida na versão 6. Pelo contrário, ainda apóio fortemente tais tecnologias, mas ganhamos um forte aliado: IntraWeb. Podem ficar tranquilos, teremos um capítulo inteiro sobre o nosso mais novo companheiro.

Aproveitando o embalo das apresentações, no próximo tópico veremos todo o potencial do Delphi 7 em todas as suas distribuições, inclusive da novíssima **Studio Architect**.

Novidades do Delphi 7

Primeiramente, irei apresentar as mudanças na nova versão do Delphi e, em seguida, uma tabela comparativa entre as suas distribuições.

- Nas mensagens de compilação, poderemos obter novas informações através do site da Borland, com um link direto da mensagem.
- Maior controle sobre os famosos “warnings” ou “advertências” que são geradas.
- Uma nova seção System para aplicações CLX, com diversos controles de diretórios e arquivos. Anteriormente tais controles estavam disponíveis somente para aplicações VCL.
- Novas implementações da Nevrona, com melhorias no pacote Indy Components, trazendo Indy Intercepts e Indy I/O Handlers, facilitando ainda mais o desenvolvimento de aplicações para Internet.
- O poderosíssimo IntraWeb, com as seções IW Standard, IW Data, IW Client Side e IW Control.
- Para quem reclamava do QuickReport, não pode mais chorar. O Delphi 7 traz consigo o poderoso Rave Reports, com superioridade absoluta em relação ao QR.
- A cada nova versão, a equipe de desenvolvimento melhora o Code Insight. Agora o **code completion** (uns dos fortes atrativos do Code Insight) está bem mais rápido, e com inteligência artificial (este é por minha conta). Sem sombra de dúvidas ficou bem mais esperto e rápido.

- Temos também um **code completion** para código HTML.
 - Você também poderá criar seus próprios administradores **code completion**, através da OpenTools.
 - Para quem não dispensa as **Watch Lists**, irá babar com as melhorias: múltiplas seções para organizar suas watches (seus vigilantes, observadores) através de grupos. Controle completo sobre a visão das watches.
 - Possibilidade de compilar projetos por grupo através do **Project Manager**.
 - A seção **Message View** foi dividida em diversas abas para apresentar diferentes tipos de mensagens (Build, Search, e outros).
 - Agora podemos selecionar múltiplos componentes na opção **View/Component List**.
 - O editor suporta diferentes tipos de linguagem: Pascal, C++, C#, HTML e XML.
 - Possibilidade de visualizar graficamente “tabs” e “espaços” no editor de código.
 - Melhorias para o desenvolvedor Web, com a inclusão do IntraWeb (perceberam como estou feliz, só falo nele), suporte ao Apache2 (com todas as suas melhorias e recursos) e exclusão do desenvolvimento de aplicações Win-CGI (16 bits).
 - Um novo browser UDDI para WebServices, e também novas classes e interfaces para o SOAP. TSoapAttachments, a grande novidade, onde podemos enviar através do SOAP, mensagens com anexo, utilizando **multipart form**. Tipos de definições são registrados automaticamente.
- Agora temos um evento poderoso: **THTTPReqResp**, onde podemos monitorar o pacote de mensagens, enquanto as mesmas são transmitidas.
- Novos drivers para o dbExpress, trazendo Informix SE, Oracle 9i, DB2 7.2, Interbase 6.5, MySQL 3.23.49, MSSQL-2000 (uma das grandes novidades).
 - A Borland desaprova o uso dos SQLLinks, e não disponibiliza mais suporte nas próximas versões do Delphi (após 2002). Ela recomenda a utilização da tecnologia dbExpress, que sinceramente, vem melhorando a cada versão.
 - Suporte a Windows XP Themes nas versões Professional e Enterprise.
 - A nova unit **DBClientActns**, com 3 novas classes: *TClientDataSetApply*, *TClientDataSetUndo*, *TClientDataSetRevert*.
 - E os novos componentes dbExpress: *TSimpleDataSet* para aplicações simples e de 2-camadas (*TSimpleDataSet* substitui *TSQLClientDataSet*).
 - Diversos componentes visuais e de diálogo para CLX.
 - Melhorias na unit **Math**.
 - Model-Maker. Para quem nunca ouviu falar em UML, sugiro conhecer esta importante especificação. No Delphi 7, a partir da versão Studio Enterprise, temos uma excelente ferramenta específica para integração UML-Delphi-UML.
 - E muito mais.

Para que você tenha uma idéia de qual versão adquirir, a Borland disponibiliza a sua tabela comparativa. Resolvi adaptá-la, fazendo breves comentários, para que você tenha uma idéia mais clara sobre as diferenças de versões. Nesta tabela são apresentadas somente as novidades do produto. Para maiores informações, sugiro uma visita ao site da Borland Latin América: www.borland.com.br

Tabela de características .NET

Características	Studio Architect	Studio Enterprise	Studio Professional	Personal Edition
. NET (Interoperabilidade e suporte a migração)				
Compilador compatível com .NET, abrangendo “warnings” e “hints”	☺	☺	☺	☺
Importa qualquer objeto .COM no padrão .NET	☺	☺	☺	☺
Exporta objetos .COM desenvolvidos em Delphi para aplicações .NET	☺	☺	☺	☺

. Delphi™ para Microsoft® .NET prerelease preview				
.NET preview compilador CIL para linguagem Delphi	😊	😊	😊	😊
Migração para documentação .NET	😊	😊	😊	😊



Minha opinião

A plataforma .NET foi bem planejada e possui uma estrutura bem diferente do que estamos acostumados. A velha “briga” entre desenvolvedores Delphi e Visual Basic deve ser deixada de lado. Acho que todos devem amadurecer e optar pela ferramenta mais adequada aos seus projetos. A Microsoft está pensando desta maneira, o que é muito saudável para todos nós. Para terem uma idéia do que estou querendo dizer, a Borland sempre suportou em seus produtos tecnologias de ponta, como Java, Object Pascal, C++ e agora .NET.

Tabela de características Model Maker

Características	Studio Architect	Studio Enterprise	Studio Professional	Personal Edition
Tecnologia Model Maker				
Design modelagem dirigida	😊	😊		
Modelagem Visual, baseada em UML	😊	😊		
Engenharia reversa com visualização instantânea, integrada ao Delphi	😊	😊		



Minha opinião

Atualmente estamos vivendo uma fase bem madura de desenvolvimento de software. Antigamente tínhamos aqueles “fantasmagóricos” fluxogramas, passando pela modelagem de dados, e agora, três vivas: UML (Unified Modeling Language, ou Linguagem de Modelagem Unificada). Ao contrário dos outros métodos, a UML é uma linguagem de modelagem muito usual em empresas fabricantes de software. E o mais incrível, foi estendida para outros tipos de segmentos, como automóveis, hardwares, empresas de energia, enfim, uma grande variedade.

O Model Maker facilita muito o trabalho com UML, integrando totalmente as classes e objetos do Delphi em seu editor.

Sugiro a todos que se aprofundem no estudo da UML. Além dos fortes benefícios, está se tornando uma grande exigência para desenvolvedores, analistas e coordenadores de projetos, além, é claro, dos gerentes.

Tabela de características IntraWeb

Características	Studio Architect	Studio Enterprise	Studio Professional	Personal Edition
Tecnologia IntraWeb (AtoZed)				
Desenvolvimento de aplicações WEB utilizando recursos visuais no estilo drag-and-drop.	☺	☺	☺	
Adicione conteúdo interativo para seus sites, de maneira rápida e visual	☺	☺	☺	
Controle transparente de cookies e sessions	☺	☺		
Grande variedade de componentes visuais para Web	☺	☺	☺	

**Minha opinião**

Quando tive meu primeiro contato com o IntraWeb pensei: essa é a grande solução para o maior problema: **prazo**. Amigos, sem sombra de dúvidas, o IntraWeb otimiza diversas tarefas no desenvolvimento de aplicações Web. Até mesmo quem não entende nada de HTML, JavaScript, desenvolve poderosas aplicações Web, com uma interface de primeira.

Fico feliz em ter o meu sonho realizado, de aumentar em mais de 30% a produtividade de minhas equipes. E o melhor, não requer um grande treinamento.

Com o capítulo dedicado a esta incrível ferramenta, tenho certeza de que todos vocês estarão desenvolvendo aplicações para Internet com extrema facilidade.

Outro ponto bastante importante é que a Borland fechou diversas parcerias para o seu novo produto, sempre pensando em disponibilizar o que há de melhor no mercado para nós desenvolvedores.

Tabela de características Nevrona Rave Reports

Características	Studio Architect	Studio Enterprise	Studio Professional	Personal Edition
Rave Report versão Delphi				
Poderoso gerador de relatórios Rave, com desenho visual de relatórios e código baseado em API	☺	☺	☺	
Gera PDF, HTML, RTF e formato texto	☺	☺	☺	
Suporte nativo para VCL e CLX	☺	☺	☺	
Estilo flexível de layout	☺	☺	☺	

**Minha opinião**

Amigos, alguém lembra do QuickReport? Eu já esqueci, nem sei do que estamos falando!

Brincadeiras à parte, finalmente ganhamos um gerador de relatórios bastante profissional. A nossa Mãe-Borland conseguiu quebrar algumas barreiras para que esse sonho fosse realizado, e todos nós ganhamos com isso. O Rave Reports da inovadora e genial empresa Nevrona (a mesma criadora do Indy Componentes) é muito estável e traz diversos recursos que facilitam a vida do desenvolvedor.

Sugiro que criem coragem de aprender um novo gerador de relatórios. Digo isso porque ao longo de minha carreira presenciei diversas pessoas e equipes que relutaram em adotar um novo gerador de relatórios, por diversas crenças: “Será que a empresa que produz o software vai manter a compatibilidade? Será que a empresa não vai quebrar? Acho muito difícil este software!” Entre outras...

Tabela de características Bold for Delphi

Características	Studio Architect	Studio Enterprise	Studio Professional	Personal Edition
Bold versão Delphi (BoldSoft)				
Assegura menor código utilizando o padrão MDA (Model Driven Architecture)	☺			
UML. Suporte total para diagramas de classes, valores identificados, entre outros	☺			
Integração com Rational Rose® e Model Maker®.	☺			

**Minha opinião**

Continuando com o nosso amigo UML, o conjunto de ferramentas da BoldSoft vem para complementar o suporte do Model Maker.

Através de componentes, o desenvolvedor poderá gerenciar de maneira bastante profissional seus diagramas UML.

Como disse anteriormente, UML é um tema muito exigido ultimamente, e recomendo um profundo estudo sobre este maravilhoso “mundo”.

IDE

A IDE do Delphi 7 (figura 1.1) é muito parecida com a versão anterior. Basicamente mudou o seu visual, muito semelhante ao Windows XP. Vamos dar uma rápida olhadinha.

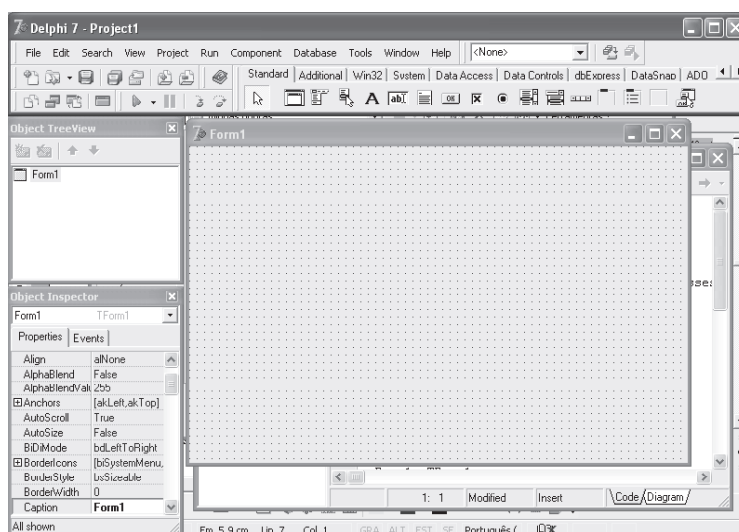


Figura 1.1 IDE Delphi 7

Como este livro está voltado para o desenvolvimento de aplicação para Internet, vamos conhecer algumas novidades.

A figura 1.2 ilustra a nova seção de objetos *Web Documents*. Sua principal função é de auxiliar na criação dos principais tipos de documentos Web.

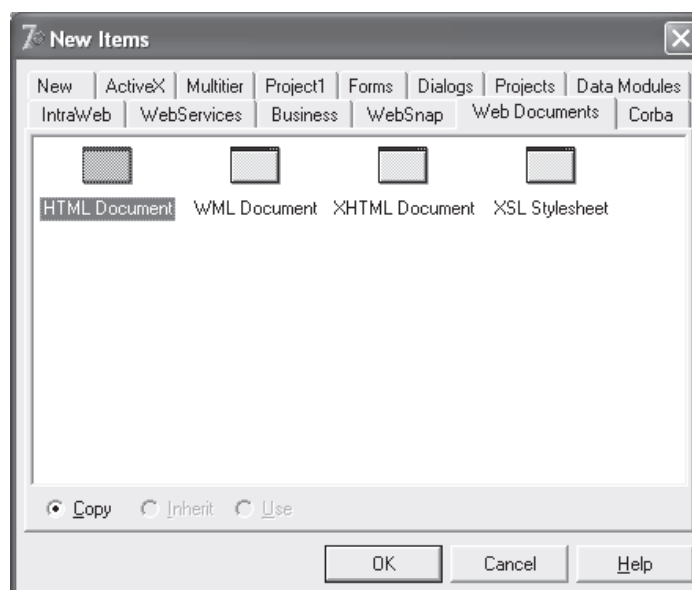


Figura 1.2 Nova seção Web Documents

Outra grande novidade é o nosso amigo *Intraweb* (figuras 1.3, 1.4, 1.5, 1.6 e 1.7) que ganhou um capítulo inteiro neste livro.

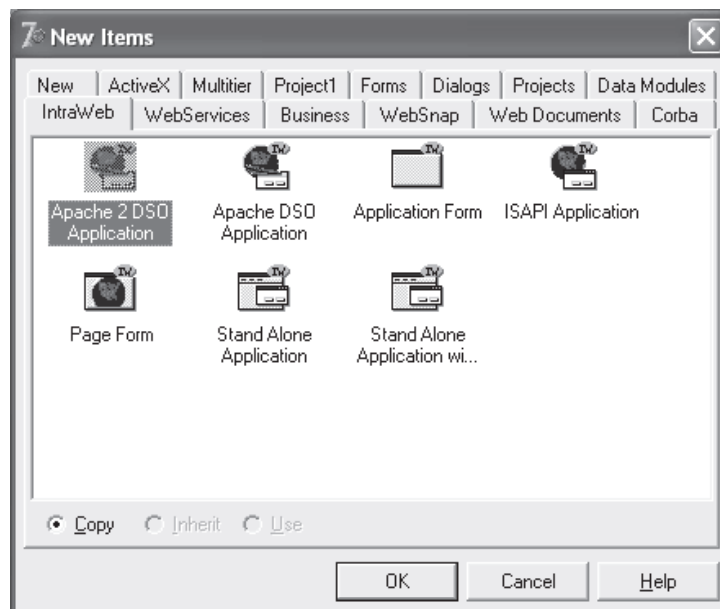


Figura 1.3 Seção IntraWeb

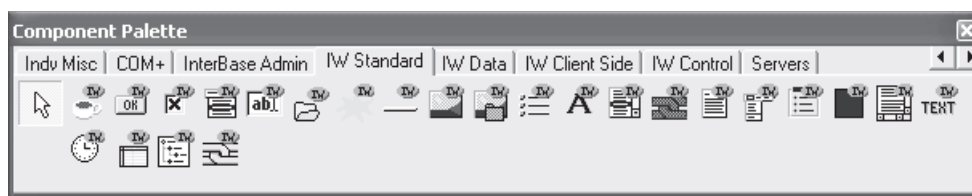


Figura 1.4 Paleta IWStandard (IntraWeb)

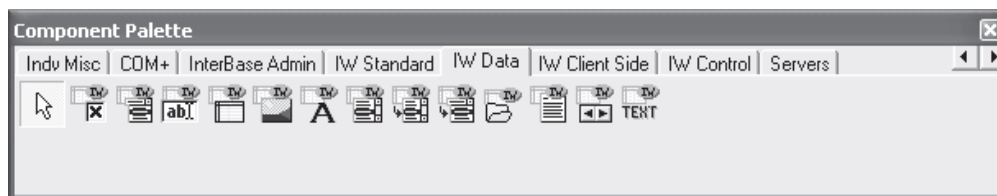


Figura 1.5 Paleta IWData (IntraWeb)

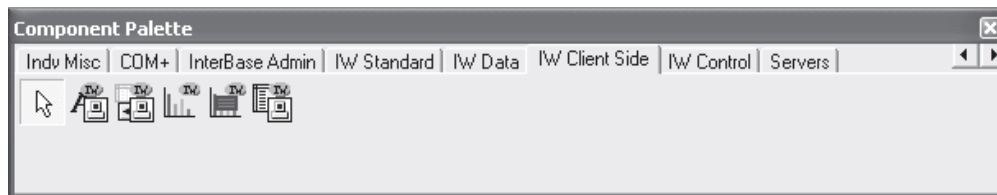


Figura 1.6 Paleta IW Client Side (IntraWeb)

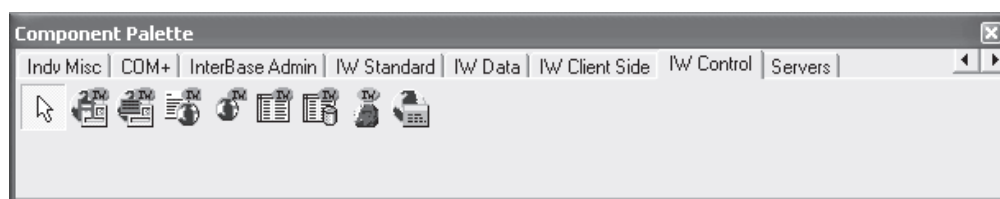


Figura 1.7 Paleta IW Control (IntraWeb)

Amigos, dá pra imaginar o poder do IntraWeb, com mais de 50 componentes nativos, no padrão CLX, só para o desenvolvimento de aplicações Web? Vocês devem estar pensando: nossa, eu nem aprendi tudo o que o Delphi me oferece de recursos e os caras disponibilizam mais poder, mais novidades! Será que eu vou aprender tudo isso? Eis a questão. Por isso um livro especializado em Internet, focado no desenvolvimento de aplicações e-Business. E o já consagrado WebSnap? Uma das estrelas da versão 6, ganhou algumas melhorias e as devidas correções dos pequenos bugs. A *figura 1.8* ilustra os componentes do WebSnap.



Figura 1.8 Paleta WebSnap

E a moda do momento? *WebServices na veia*. A tecnologia de *WebServices* já está sendo utilizada, mesmo que timidamente, em diversos setores, e fará mais sucesso ainda com as ferramentas que vêm surgindo no mercado. A *figura 1.9* ilustra a seção de *WebServices* do Delphi 7, e a *figura 1.10*, a paleta de componentes.

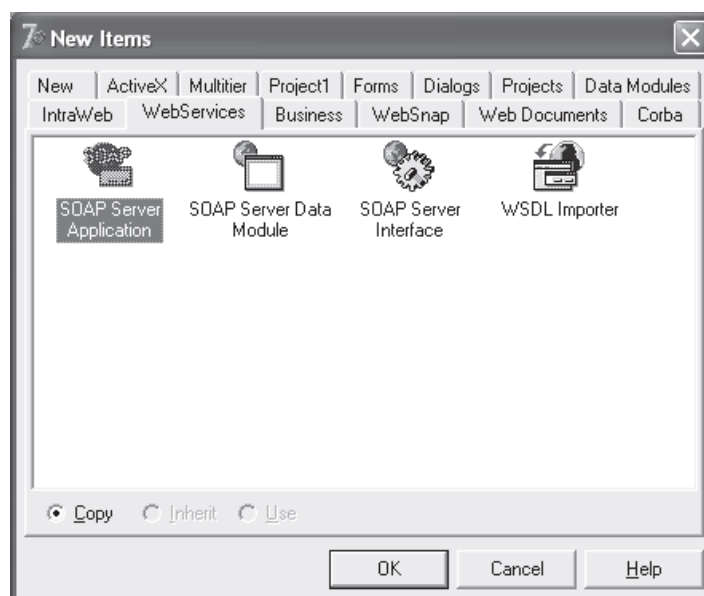


Figura 1.9 Seção WebServices

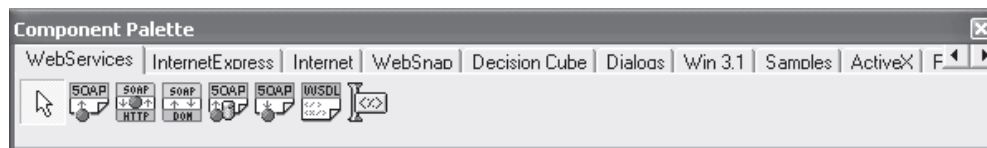


Figura 1.10 Paleta WebServices

Amigos, tivemos uma breve apresentação dos novos recursos do Delphi 7 na área de desenvolvimento para Web. A seguir teremos uma introdução bastante importante sobre o desenvolvimento de aplicações e-Business, bem como sobre o Método Facunte.

Visão geral do e-Business

No cenário atual temos a enorme necessidade de produzir novos produtos baseados na tecnologia e-business. Mas que produtos?

- ERP
- CRM
- B2B
- B2C
- entre outros

Vejamos a evolução do mercado de desenvolvimento:

MEADOS	PLATAFORMA (*)	LINGUAGENS/FERRAMENTAS
80 a 90	Unix, IBM, DOS	Cobol, Dataflex, C, C++, Clipper
91 a 94	DOS, Windows	DataFlex, Clipper, Fox, Visual Basic
94 a 97	DOS, Windows	Clipper, Visual Basic, Delphi
98 e 99	Windows, Internet	Visual Basic, Delphi , ASP, PHP
00 a 01	Windows, Linux, Internet, Intranet, Extranet	Visual Basic, Delphi , Kylix , ASP, PHP, ColdFusion, Java
02 tendências	Cross-plataform, objetos distribuídos	. NET, Delphi , Kylix , Corba (ORB) , Java , SOAP , SNAP (Web Services)

TABELA 1 EVOLUÇÃO DO MERCADO DE DESENVOLVIMENTO
(*) Principais tecnologias

Como observamos na **Tabela 1**, o mercado evoluiu muito após a década de 90, tornando confusa a decisão do desenvolvedor na escolha da melhor tecnologia. Para ajudar os desenvolvedores da “velha-guarda”, bem como os da “nova-geração”, foi criado o **MÉTODO FACUNTE** (falaremos do método em seguida). Bem, **Método** à parte, falaremos da forte tendência para esta década que, conforme a **Tabela 1**, é o desenvolvimento Cross-Plataform e Objetos Distribuídos. Mas o que são estas novas tecnologias? Será que são novas mesmo?

Vamos iniciar pela tendência Cross-Plataform. Cross-Plataform quer dizer que a aplicação “roda” em duas ou mais plataformas. Isso é excelente, já que estamos num mercado bastante disputado; Windows melhorando, Linux cativando grandes legiões de “adeptos”, e quem sabe o Lindows¹ avança no mercado. Agora vamos imaginar nossa aplicação sendo executada em qualquer ambiente operacional. Não é ótimo? Já pensou em oferecer ao seu cliente um produto Cross-Plataform? Tenho certeza de que aumentarão muito as chances de vender o produto.

OK, e os Objetos-Distribuídos? Bem, esse é o ponto. Vamos imaginar o seguinte cenário:

- Aplicação de Controle de Estoque sendo acessada pela LOJA.
- Faturamento acessando informações de Vendas.

¹ Lindows – Sistema operacional baseado no LINUX, que traz consigo a tecnologia WINE (kernel para execução de aplicações baseadas em Win32).

- Financeiro gerenciando Faturamento e Contas a Pagar.
- Diretoria gerenciando todo o sistema.
- Clientes consultando e fechando pedidos através da Internet.
- Setor de compras fechando pedidos com Fornecedores.
- Compensação automática de crédito/débito da Rede Bancária.
- Vendedores externos fechando negócios com PALM TOPS.

Sem dúvida é um sistema dos sonhos de qualquer empresário e desenvolvedor. Agora imagine efetuar uma manutenção no sistema sem interromper as atividades normais (não vá pensar em atualizar o sistema as 02:00 da manhã, hein?! Isso não é tudo). Vamos melhorar o nosso “lado”?

Com a tecnologia de Objetos-Distribuídos podemos criar diversas camadas:

- Camada de Negócios
- Camada da Aplicação Cliente
- Camada da Aplicação Servidor
- Camada Aplicação Servidor-WEB
- Camada Banco de Dados
- entre outras

Para realizar as tarefas de manutenção no sistema, o desenvolvedor poderá alterar apenas a camada que satisfaz a ocasião. Outro fator bastante interessante é que as camadas podem e devem ficar em servidores diferentes, específicos para cada camada.

Tudo bem Facunte, mas o que a tecnologia de desenvolvimento WEB tem a ver com **tudo** isso? Isso mesmo: **TUDO!** Bem, quando nos referimos a Cross-Plataform imaginamos que os nossos clientes pudessem executar a mesma aplicação em diversas plataformas, certo? Então temos aqui um caso típico de Cross-Plataform – tudo bem, no nível de Client e não Server – eu explico:

Desenvolver uma aplicação com a tecnologia WEB utilizando o Delphi, a mesma poderia ser executada num servidor Windows NT e qualquer equipamento (micro-computador, PALM, Celular, etc. acessaria a aplicação, mesmo que o sistema operacional não fosse Win X. Exemplo: Linux, Mac OS, Solaris, celulares com browser baseado em JAVA (esqueça WAP por enquanto), PALM com acesso à Internet, entre outros. Viram que magnífico?

Em resumo, o objetivo deste livro é o desenvolvimento de uma **Camada Web**, uma das grandes tendências desta década.

Método FACUNTE

O Método Facunte foi criado em 2001 inspirado nas necessidades de desenvolvedores iniciantes e dos amigos da “velha-guarda” que não tinham base formada sobre qual tecnologia adotar.

O método consiste num treinamento completo para o desenvolvedor, abrangendo desde as tendências do mercado, passando por orientações financeiras de projeto, até o desenvolvimento prático de uma aplicação, utilizando ferramentas de alta tecnologia e comumente aceitas como padrão de mercado.

No diagrama a seguir temos a representação gráfica do método.



Aspectos Gerais para o Desenvolvimento de Aplicações Web

O desenvolvimento de aplicações para web difere do método tradicional em alguns pontos:

- Aplicações são executadas no browser (Internet Explorer, Netscape, etc.);
- Para construir formulários utilizamos HTML;
- Integração com diversas tecnologias: Flash, JavaScript, Java.

Devemos esquecer alguns costumes do método tradicional, como por exemplo:

- Grids;
- Barras de navegação entre registros;
- Botões de funções para todos os lados;
- Sobreposição excessiva de janelas.

Os referidos costumes degradam muito a performance do servidor de aplicações e de banco de dados, além de confundir o usuário em alguns aspectos.

Outra coisa bastante interessante é que a maioria dos desenvolvedores quer trazer para a web uma interface parecida com seu sistema tradicional. Isso pode custar “caro”, pois exige um esforço grande de desenvolvimento.

Para tanto, recomendo um novo conceito de interface, que veremos ao longo do livro. Vejam que isso é apenas um exemplo, que na minha opinião deve ser seguido, mas novos conceitos deverão ser criados, e até mesmo outros tipos de interfaces existentes no mercado devem ser analisadas. Procure sempre o melhor para o usuário.

Como está o mercado de trabalho

Desenvolvedores de aplicações para web, ou melhor, e-business developers, estão muito bem cotados no mercado atual. Basta olhar para empresas como Oracle, Siebel, Jd Edwards, SAP, People Software, que estão criando ou migrando seus produtos para a tecnologia WEB. Veja a média de salários para desenvolvedores WEB.

cargo	desenvolvedor tradicional	desenvolvedor web
desenvolvedor jr	R\$ 800 a R\$ 1200	R\$ 1200 a R\$ 1800
desenvolvedor pleno	R\$ 1000 a R\$ 2000	R\$ 1800 a R\$ 3500
desenvolvedor sênior	R\$ 2000 a R\$ 4000	R\$ 3000 a R\$ 8000
Gerente de projetos	R\$ 2000 a R\$ 6000	R\$ 4000 a R\$ 12000
arquiteto web	-	R\$ 6000 a R\$ 10000
consultor	R\$ 20 a R\$ 50/hora	R\$ 40 a R\$ 130/hora

*TABELA 2 pesquisa salarial realizada nos principais sites de RH
(Grupo Catho, AP Info, Canal de Empregos, Manager)*

A demanda de desenvolvedores web ainda é muito pequena, e a oferta de trabalho é muito grande, por isso temos esse cenário. Claro que aqueles que entrarem antes no mercado serão altamente beneficiados. Não se assustem com as quebraadeiras das empresas *pontocom*, pois este tipo de trabalho não se restringe a este seguimento. O que estamos mostrando aqui é que qualquer empresa pode contratar os serviços de um desenvolvedor web, desde a lojinha de autopeças da esquina, até as grandes corporações.

Como definir os custos de um projeto

Quantas e quantas vezes nós desenvolvedores falamos a seguinte frase: **“Puxa, deveria ter cobrado mais, este projeto está me causando fortes dores de cabeça”**. Acredito que inúmeras vezes, não é? Sem sombra de dúvidas temos o vilão da estória: **péssimo planejamento**.

Então como devemos planejar e definir os custos de um projeto? Bem, ainda não inventaram fórmulas mágicas, mas estão chegando perto disso. Mas enquanto não descobrem, vamos checar algumas sugestões:

- **Reflita sobre a sua experiência** – por exemplo: um cliente solicita um orçamento para o desenvolvimento de uma aplicação de controle de estoque. Você já desenvolveu alguma aplicação deste tipo?
NÃO – então aumente relevantemente o tempo de desenvolvimento.
SIM – bem, com isso você sairá na frente, reduzindo assim o tempo de desenvolvimento do projeto.
- **Pesquisar o máximo junto ao cliente** – antes de emitir a proposta, pesquise o máximo de informações junto ao cliente, para saber até onde ele imagina que a aplicação vai chegar. Não são raros os casos de clientes que pedem uma simples aplicação de controle de estoque e depois questionam: “onde está o meu contas a pagar?” E o desenvolvedor retruca: “você não pediu isso!” Não é mesmo, amigos? Isso faz com que o desenvolvedor tome uma das seguintes atitudes:
 - desenvolve o módulo e não cobra nem um centavo a mais
 - tenta convencer o cliente de pagar uma quantia extra pelo desenvolvimento (isso causa fortes dores de cabeça)
 - “chuta o balde” e deixa o cliente “na mão” (nunca façam isso, amigos.)
 - entre outras coisas
- **Definir o prazo junto ao cliente** – quanto menor o prazo, maior será seu custo, pois deverá mobilizar uma equipe ou até mesmo atravessar noites em claro.

- **Nunca faça leilão** – é comum o seu cliente querer reduzir o valor de sua proposta, apresentando outras de seus concorrentes. Casos absurdos de redução de 50, 60 e até 70% são comuns. Acredito num valor de redução máximo de 20% e o ideal em 10%.

Tudo bem, Facunte, mas como definir os custos de um projeto? Minha sugestão é a seguinte:

- Defina seu valor hora, entre R\$ 40 e R\$ 60 (sessenta reais).
- Baseado em sua experiência, calcule o tempo necessário para desenvolver a aplicação, em número de horas e multiplique por 2,5.
- Caso haja a necessidade de outros membros no desenvolvimento do projeto, faça a adequação em número de horas e acrescente 20% por membro.
- Acrescente os impostos.
- E “finito”, encaminhe a proposta ao cliente.

Veja um exemplo prático:

Minha estimativa em horas do projeto	: 500 horas
Tempo necessário	: 500 horas x 2,5 = 1.250 h
Prazo estimado pelo cliente	: 2 meses
O prazo fornecido pelo cliente forçou-me a	
Introduzir mais 3 membros na equipe	: 1.250h + 60% = 2.000 h (60% = 20% por membro)
Valor hora	: R\$ 40,00
Valor total do projeto	: 2.000 x R\$ 40,00 = R\$ 80.000,00

Repare, o que “pesou” no valor final do projeto foi o prazo estipulado pelo cliente, senão teríamos um valor bem mais atraente, em torno de R\$ 45.000,00. Lembre-se de estimar as horas baseadas em sua experiência, e não façam leilões com o seu valor hora.

Obviamente numa “fábrica de software” – virou moda este nome – são utilizados recursos mais avançados para avaliação de valores.

Como gerenciar uma equipe de desenvolvimento

Bem amigos, o que irei explanar aqui é apenas uma base de como gerenciar **pequenas** equipes de desenvolvimento, pois para uma idéia mais abrangente seria necessário escrever um bom livro.

Em primeiro lugar deve-se definir o gerente do projeto. Nunca pense que por estar entre amigos o gerente irá deixar de dar as ordens ou os caminhos para o desenvolvimento do projeto. Para gerenciar uma equipe é necessário liderança, e quem não nasce com o espírito de líder deverá conquistá-lo.

O gerente deverá definir os responsáveis por cada tarefa, além de regras básicas para que o projeto possa fluir tranquilamente:

- responsável geral (gerente)
- responsável pelas regras de negócio
- coordenador do projeto
- responsável pela modelagem de dados
- responsável pelo desenvolvimento de classes básicas
- responsável pelo design do projeto
- responsável pelo núcleo do projeto
- responsável pelos testes

Costumo dizer que, após a modelagem de dados, desenvolver fica fácil. Digo isso porque o coração da aplicação é a modelagem. Uma boa modelagem é igual a uma boa aplicação, já uma razoável modelagem é igual a uma semente frutífera de

problemas. Com a modelagem em mãos, o gerente poderá iniciar os trabalhos com cadastros básicos e design, lembrando que a modelagem de dados deve ser baseada na análise de negócios. Os testes deverão ser iniciados paralelamente ao desenvolvimento dos cadastros básicos, assim como o núcleo do projeto (caso haja um responsável por esta área, senão, somente após a conclusão dos cadastros básicos). O próximo tópico complementa as dicas sobre gerenciamento de equipes.

Quais as ferramentas adequadas para a equipe

Gerenciar uma equipe não é uma tarefa fácil, mas existem no mercado ferramentas especialmente desenvolvidas para este fim.

MS Project

O Microsoft Project® é o melhor gerenciador de projetos que conheço. Com ele podemos gerenciar diversas equipes ao mesmo tempo, conseqüentemente diversos projetos (mesmo que por equipe), definir metas e analisar gráficos de desempenho por fases, projetos e equipes. O projeto é compartilhado entre os membros da equipe, facilitando assim o andamento do mesmo, mostrando as deficiências e também toda a eficiência da equipe. Recomendo para todos os tamanhos de projeto, e acredito ser indispensável para grandes projetos.

ERWin

Quem nunca ouviu falar da fantástica ferramenta de modelagem de dados **ERWin**? Hoje pertence a CA (Computer Associates) que tem investido muito para melhoria da ferramenta. O ERWin gera scripts SQL para a maioria dos bancos de dados comerciais, como Oracle, SQL Server, Interbase, Sybase, DB2, Progress, Informix, entre outros. Para pequenos desenvolvedores não recomendo, devido ao seu alto custo (CA, não fique brava comigo, mas é a dura realidade), mas para médios e grandes é indispensável.

A Squadra, empresa 100% nacional, é fabricante de uma ótima ferramenta de modelagem de dados: Dr Case. Além do preço bastante atraente, a ferramenta traz consigo ótimos recursos para o trabalho com os principais bancos de dados (recomendo para quem utiliza diversos tipos de banco de dados).

Existem produtos alternativos específicos para Interbase, como o fantástico IB Admin, da SQLLY (www.sqlly.com) com um custo bastante atraente, e com diversos recursos de gerenciamento do Interbase (altamente recomendável para quem utiliza Interbase).

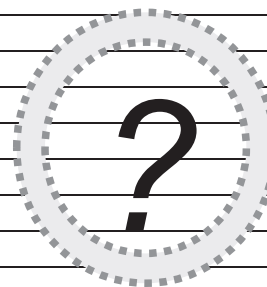
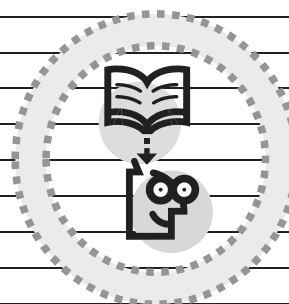
UML

Para quem quer qualidade total em seus projetos, recomendo o trabalho baseado em UML (Unified Modeling Language, ou Linguagem de Modelagem Unificada). Só para ter uma idéia, além das grandes empresas de software, estão adotando UML indústrias automobilísticas (GM, Ford), fabricantes de hardware (Intel, EPSON, HP, Compaq), entre outras. Mas o que é UML? E quais as suas vantagens? Em resumo, UML é a unificação dos métodos Booch, OMT e OOSE, padronizando a modelagem de sistemas de software orientados a objetos, entre outras aplicações. Hoje em dia, a recomendação básica para desenvolvimento de sistemas é a utilização de todo o poder da orientação a objetos. A UML colabora muito para a organização e documentação de projetos orientados a objeto. Para entender melhor o UML, recomendo a leitura do livro **Desenvolvendo Aplicações com UML** da Brasport.

Mas falar de UML sem falar de sua principal ferramenta, é realmente um deslize. Ao meu ver, não existe ferramenta melhor que o Rational Rose da empresa Rational (www.rational.com.br). Todas as especificações da UML são encontradas na ferramenta, além da integração com as principais linguagens de desenvolvimento do mercado. E como agora ganhamos o Model Maker, let's go, friends!!!

TeamSource

A Borland disponibiliza uma excelente ferramenta de controle de versões e de trabalho em equipe: a *TeamSource*. Com esta ferramenta é possível gerenciar inúmeras versões do projeto, além de organizar o desenvolvimento em grupo, permitindo que inúmeros desenvolvedores participem do mesmo projeto. Maiores informações www.borland.com.br

Anotações de Dúvidas**Preciso Revisar****Anotações Gerais**