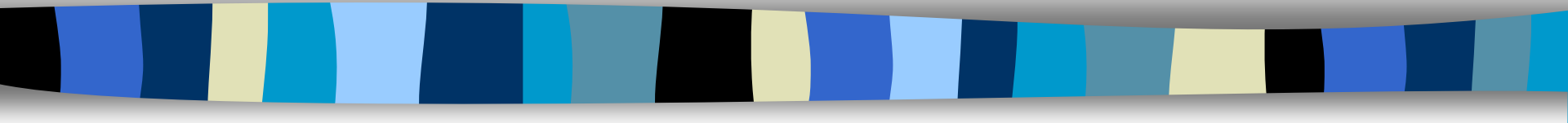


Progress ACE





Detalhes

- O que é A . C . E. ?
 - Ambiente para construção de aplicações com o Progress 8.
- O que é A . D . M. ?
 - É uma estratégia para rápido desenvolvimento, reusabilidade e manutenção de objetos e componentes.
- Qual a importância?
 - É um novo modo de desenhar e construir aplicações.

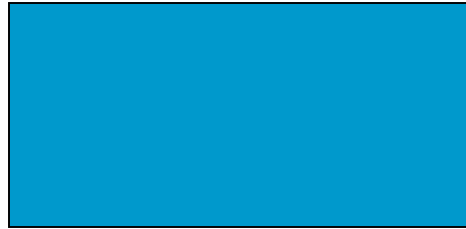


Arquitetura ADM - Introdução

- É um modelo baseado em componentes pré construídos que poderão ser reutilizados em outras aplicações.
- A comunicação entre os componentes é conhecida como “links”.
- A comunicação entre SMARTOBJECTS é conhecida como mensagem.

Componentes

SmartWindow



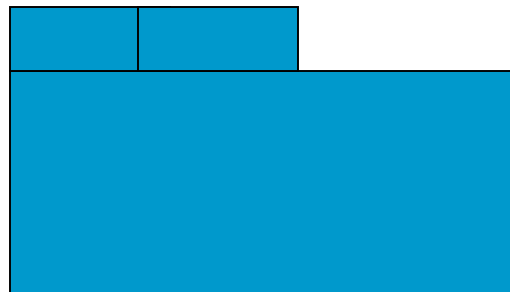
SmartFrame



SmartDialog



SmartFolder



Componentes (cont.)

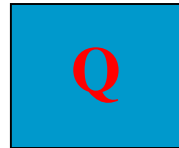
SmartPanel (navigation)



SmartPanel (update)



SmartQuery

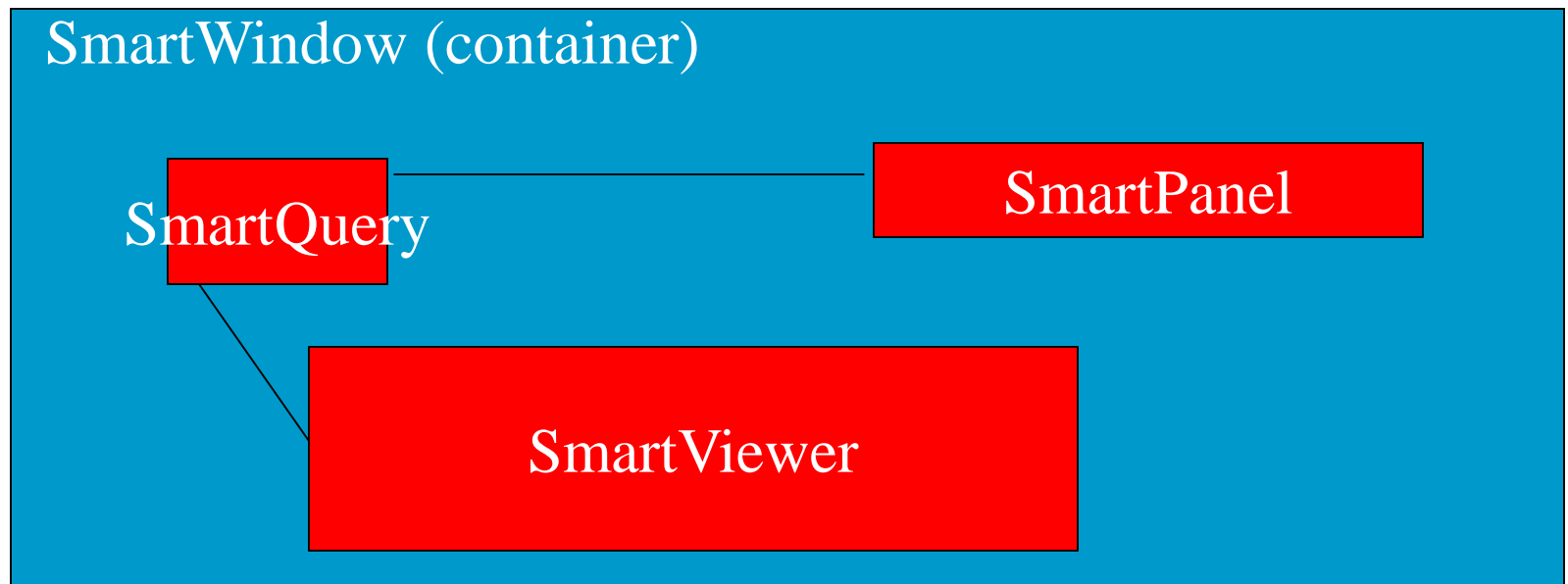


SmartBrowser



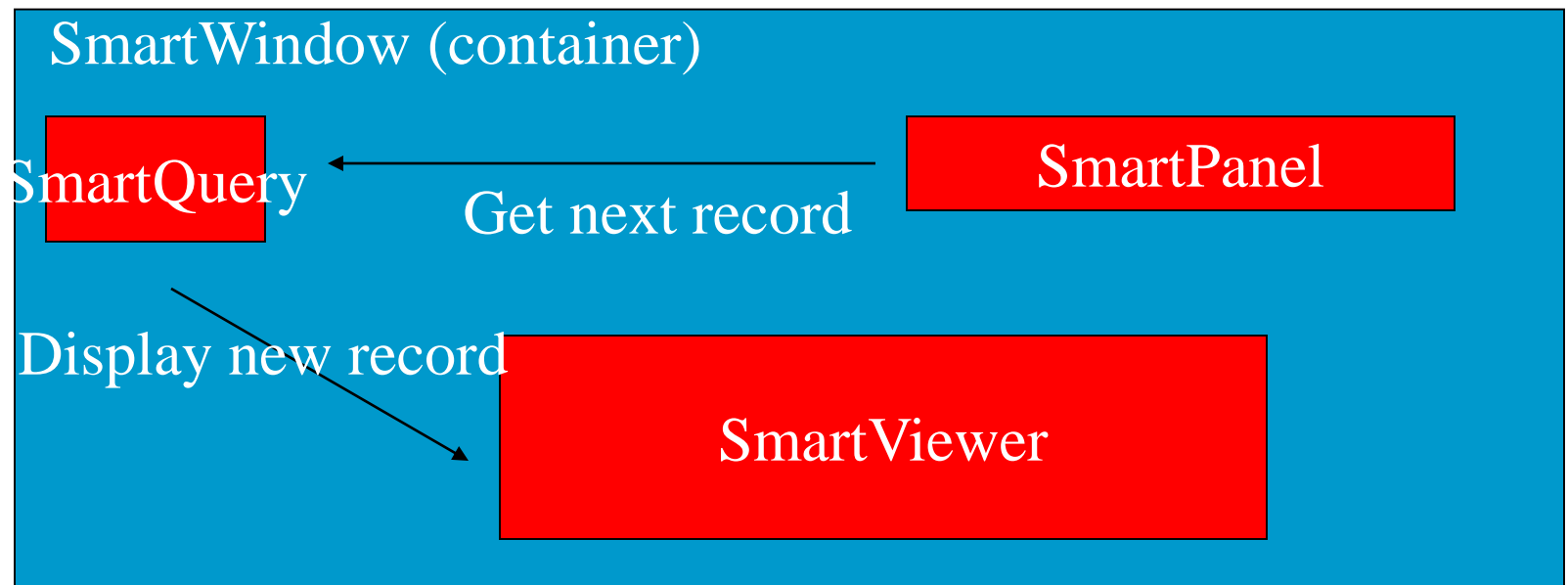
Arquitetura ADM

- Container
 - A função básica é prover o depósito de SmartObjects.
- Tipos de Containers
 - SmartWindow, SmartDialog, SmartFrame.



Arquitetura ADM (cont.)

- Em tempo de execução os SmartObjects se comunicam através das Mensagens.
 - Exemplo: quando um usuário pressiona o botão NEXT, o SmartPanel envia uma mensagem de GET NEXT RECORD para a query.
 - A QUERY pega o próximo registro e envia uma mensagem de display new record para a SmartViewer.





Arquitetura ADM (cont.)

- A palavra SMART é relacionada com a capacidade dos SmartObjects proverem a maioria das mensagens.

Evolução do Modelo

V6
Arquivos
Include

V7.3
Templates

V8
SmartObjects



Benefícios

- Alta produtividade
 - Reusabilidade,
 - Desenvolvimento visual,
- Alta qualidade
 - Objetos pré-testados,
 - Estabelecimento de padrão no desenvolvimento.
- Desenvolvimento baseado em componentes.
- Progress 4GL.
- Tecnologia Orientada a objetos.
- Programação Visual.

Exemplos

Representantes de Vendas

Navigation buttons: [Previous] [Previous] [Next] [Next]

Sales-Rep:

Rep-Name:

Region:

Clientes

Cust-Num	Name
1	Lift Line Skiing
2	Urpon Frisbee
3	Hoops Croquet Co.
4	Go Fishing Ltd
5	Match Point Tennis
6	Fanatical Athletes

Contact:

Country:

Credit-Limit:

Discount:

Name:

Exemplos (cont.)

The screenshot shows a software window titled "Clientes com mais detalhes". It contains a list of customers with columns "Cust-Num" and "Name". The customer "5 Match Point Tennis" is selected. Below the list, there are two tabs: "Endereço" and "Financeiro". The "Financeiro" tab is active, showing fields for "Credit-Limit", "Discount", and "Terms".

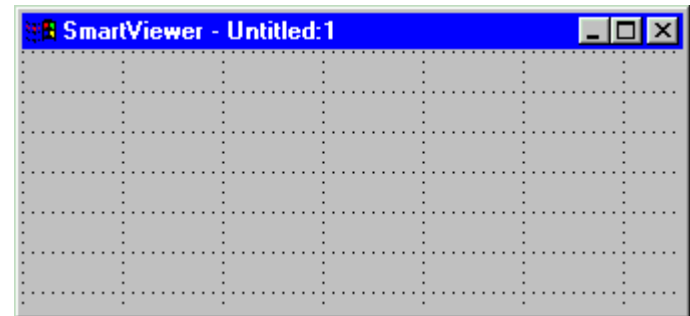
Cust-Num	Name
1	Lift Line Skiing
2	Urpon Frisbee
3	Hoops Croquet Co.
4	Go Fishing Ltd
5	Match Point Tennis
6	Fanatical Athletes

Endereço Financeiro

Credit-Limit: 11.000
Discount: 50%
Terms: Net30

Introdução aos SmartObjects

- É um componente de aplicativo reusável que interage e se comunica com outros componentes para formar um aplicação com coesão.
- SmartObject Templates:
 - Uma template é o ponto de start para a construção de um ou mais SmartObjects masters específicos.
 - Como exemplo podemos citar uma SmartViewer fornecida pela Progress, ela não contém campos da base, mas contém código nas entrelinhas para gerenciar estes campos.



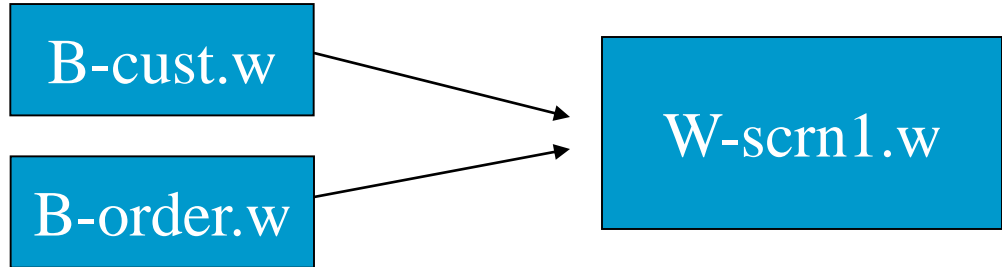
Criando Masters a partir de Templates

■ SmartObject
Template

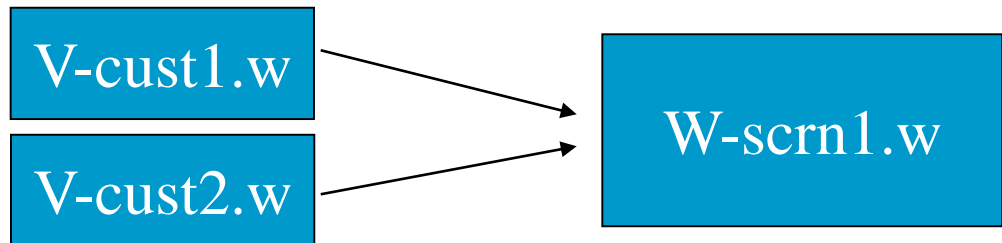
■ SmartObject
Masters

■ Aplicação

SmartBrowser



SmartViewer





Criando Masters a partir de Templates(cont.)

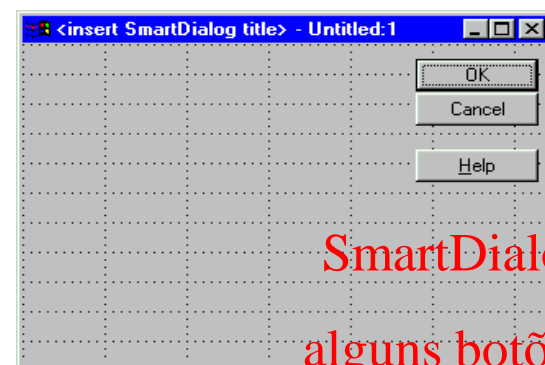
Tipos de SmartObjects

A Progress possui os seguintes SmartObjects:

- SmartContainer,
- SmartQuery,
- SmartViewer,
- SmartBrowser,
- SmartPanel.

SmartContainers

- Existem 3 (três) tipos de SmartContainers:
 - SmartWindow,
 - SmartDialog,
 - SmartFrame.



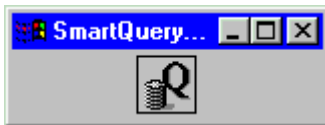
SmartDialog, inclui alguns botões padrões com código em suas triggers.



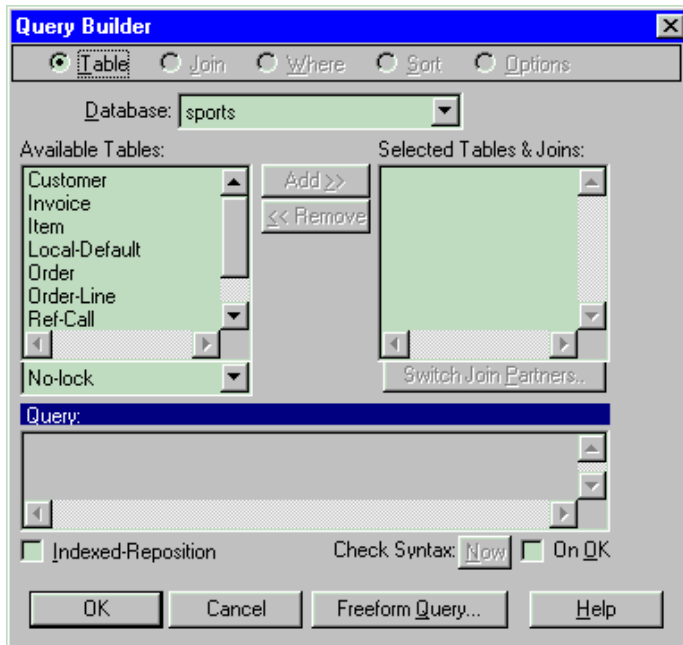
SmartContainers (cont.)

- Para que servem ?
 - Servem para gerenciar o relacionamento entre SmartObjects
- Cada SmartContainer:
 - Grava quais outros SmartObjects você inseriu,
 - Mantém o caminho do link entre os SmartObjetcs,
 - Inicializa o SmartObjects em tempo de execução.

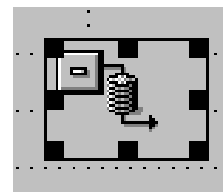
SmartQuery



Na criação de uma nova SmartQuery.



Duplo clique na
imagem para
mostrar o
Query Builder.



Quando inserido
em um
SmartContainer.

Importante! Em tempo de execução a SmartQuery NÃO possui visualização.

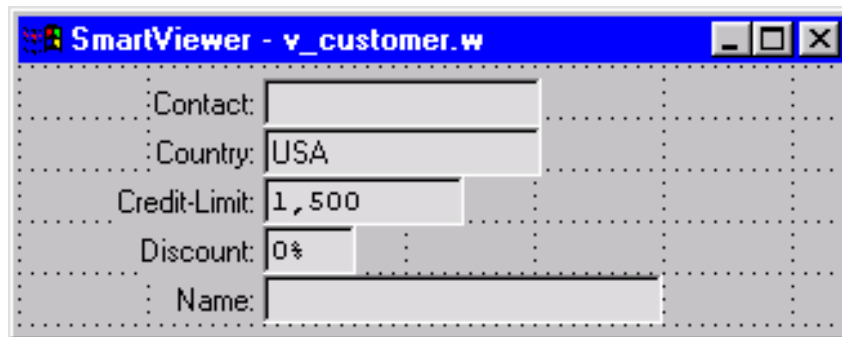
SmartQuery (cont.)

- É responsável por gerenciar um grupo de registros, identificando tabelas e campos, incluindo ainda uma cláusula where e critério para sort.
- Outros SmartObjects podem requisitar uma atualização de registros utilizando uma SmartQuery e sua capacidade de mensagens.
- Use SmartQuery como fonte de registros para o Display.
- Lembre-se que uma SmartQuery poderá ser reusável, usando-a em diferentes aplicações.
- Não se usa SmartQuery na maioria dos SmartBrowser pois estes possuem uma query em seu interior. A única exceção é quando o SmartBrowser recupera registros informados em uma query(Orders of Customers).



SmartViewer

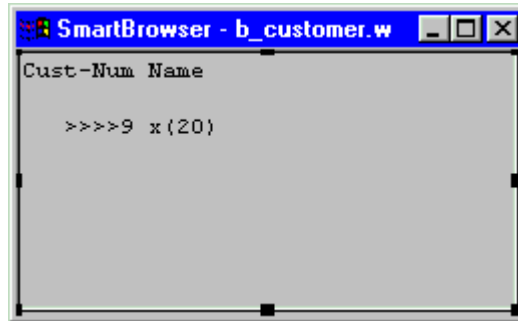
- Gerenciar um registro de uma ou mais tabelas.
- Pode-se inserir campos somente para display ou para update, add, copy, delete...
- Pode-se requisitar registros e recebê-los por outros SmartObjects através da capacidade de mensagens.
- Use SmartViewer para mostrar ou atualizar um ou mais campos. Lembrar que você deverá possuir link com um fonte de registros(SmartQuery ou SmartBrowser) para poder mostrar os dados.



The screenshot shows a window titled "SmartViewer - v_customer.w" with a standard Windows-style title bar. The main area contains a form with several input fields arranged in a grid-like structure. The fields are labeled as follows:

Contact:	<input type="text"/>
Country:	<input type="text" value="USA"/>
Credit-Limit:	<input type="text" value="1,500"/>
Discount:	<input type="text" value="0%"/>
Name:	<input type="text"/>

SmartBrowser



- É como qualquer Browser, mostra um conjunto de registros através de Scrollbar.
- Pode opcionalmente possuir campos habilitados para atualização.
- Possui internamente uma query que você define quando você constrói um SmartBrowser.
- Pode servir de fonte de registros para outros SmartObjects, podendo enviar dados para outro SmartBrowser ou uma SmartViewer para sincronizar dados entre objetos.
- Use quando quiser mostrar um grupo de registros para leitura ou para atualização.

SmartPanel

- São SmartObjects pré-compilados disponíveis para uso imediato, não se precisa criar primeiramente um master.
- Existem vários tipos de SmartPanels no Progress 8:

– Navegação,



– Update,



- Pode enviar mensagem para uma SmartQuery para a obtenção de um determinado registro, a SmartQuery então enviará os dados para o display na SmartViewer.
- Use o SmartPanel de Navegação para controlar visualmente a navegação.
- Use o SmartPanel de Update para atualizar dados em uma SmartViewer ou SmartBrowser.

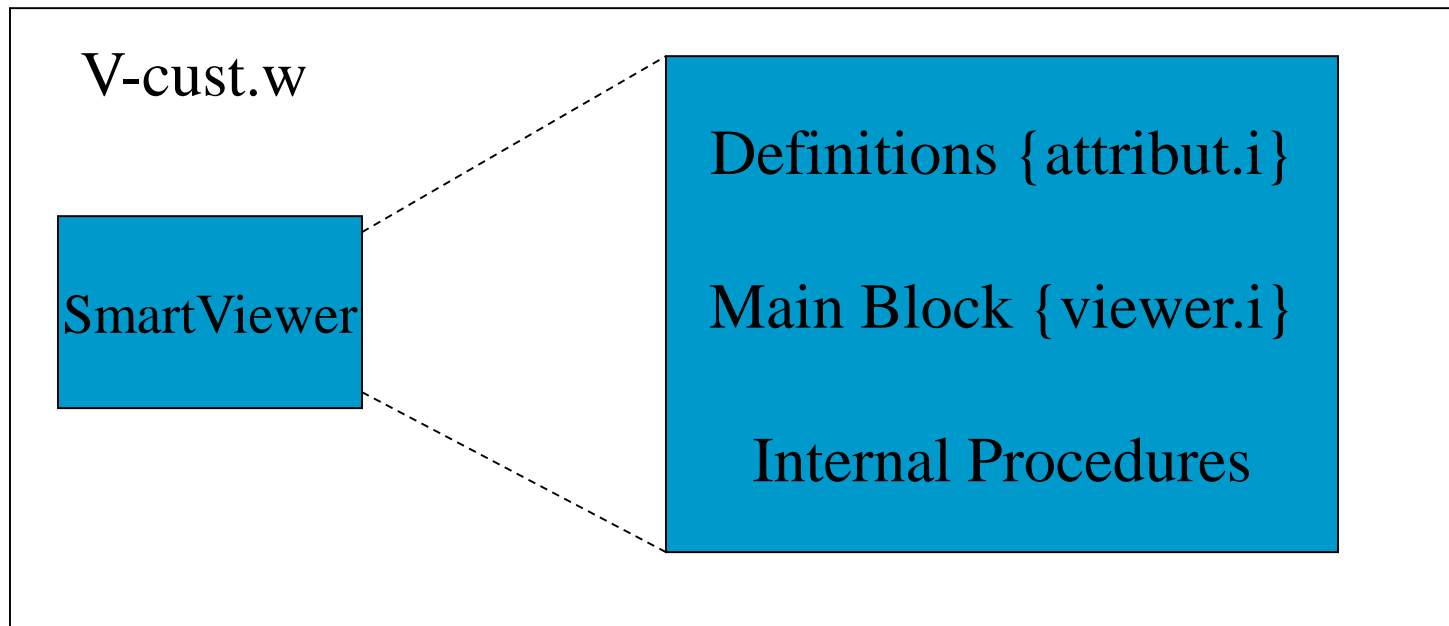


Anatomia dos SmartObjects

- SmartObjects antes de tudo são .w, possuindo sessão Main Block, Definitions, Internal Procedures.
- Pela utilização de Pré-processadores os .w são reusáveis em várias aplicações, pois o UIB gera código utilizado-os.
- Todo SmartObject possui um conjunto de procedures internas que promovem as funções padrões para aquele tipo de SmartObject, como enviar e receber mensagens, ou inicializar a tela. Os SmartObjects executam estas procedures internas em outros SmartObjects.
- A maioria das procedures estão alocadas em includes, conhecidas como biblioteca de métodos. Estas bibliotecas de métodos promovem um código reusável que permite diferentes SmartObjects trabalharem juntos.

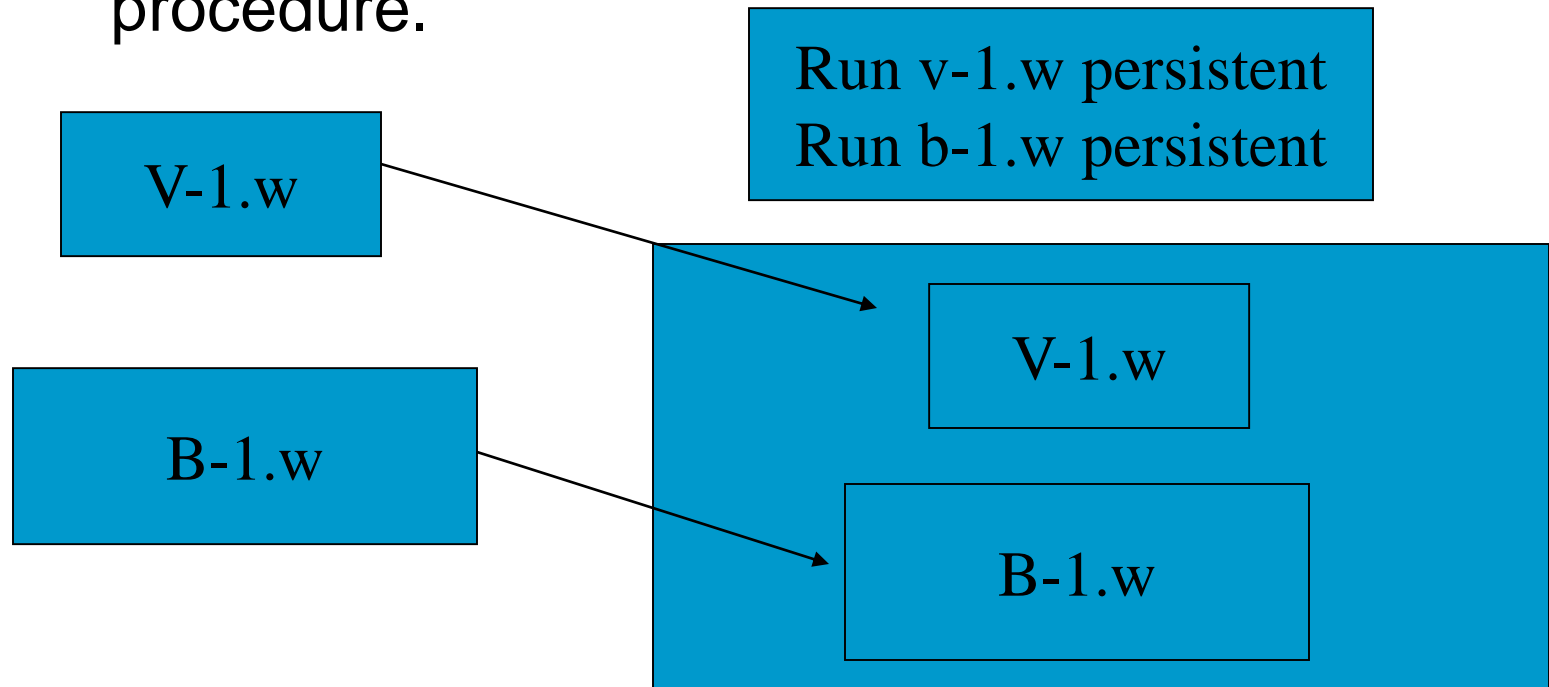
Anatomia dos SmartObjects(cont.)

- **Attribut.i** : promove definições padrões dos SmartObjects.
- **Viewer.i**: promove funções específicas e necessárias para todas as SmartViewers.
 - {smart.i} funções padrões para SmartObjects.
 - {record.i} para trabalhar com registros.
 - {tableio.i} para alterações nos registros.



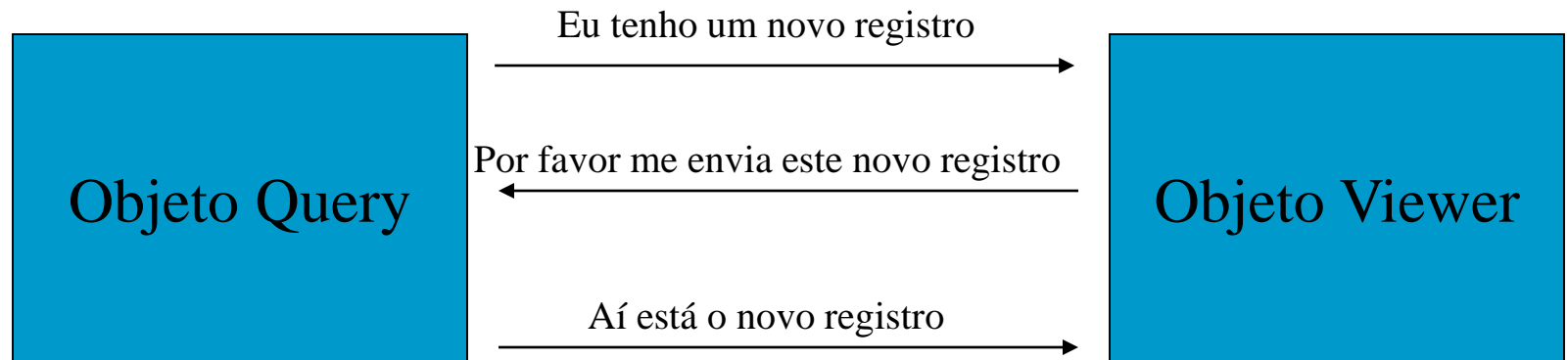
Como o ADM funciona

- Cada Objeto dentro do SmartContainer é executado independentemente através do run procedure.



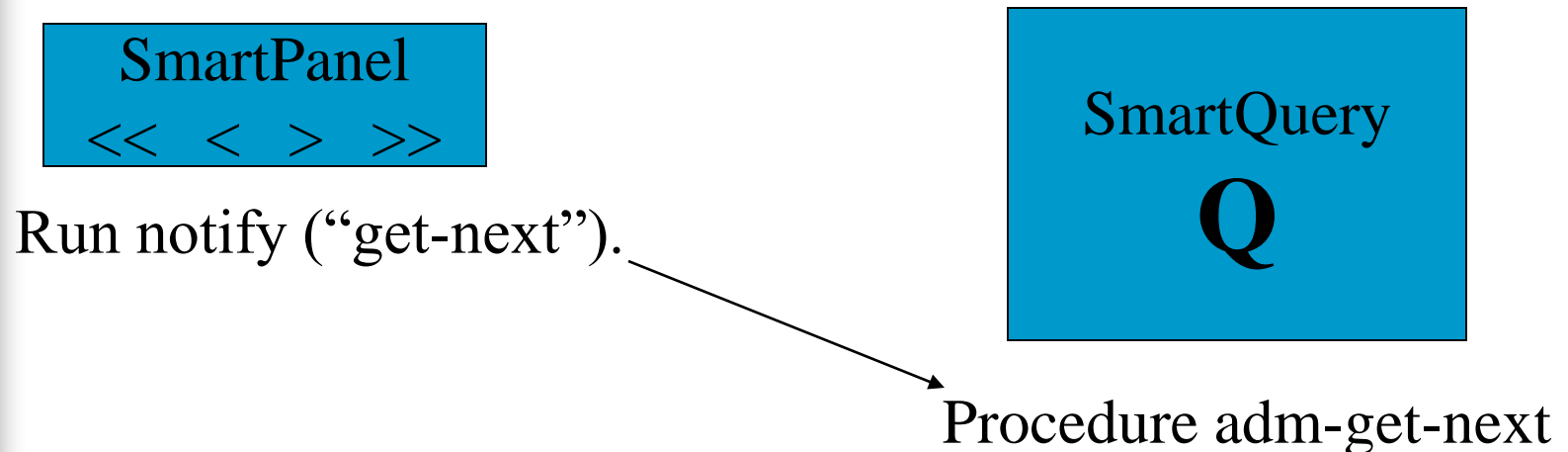
Como o ADM funciona (cont.)

- Na execução da aplicação, cada objeto se comunica com outro através de mensagens, fazendo requisição a outro objeto para executar alguma tarefa.



Como o ADM funciona (cont.)

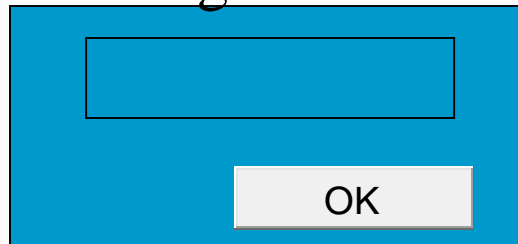
- Os objetos executam os próprios processos.
- Quando um objeto recebe uma requisição para algum processamento, o objeto executa uma procedure interna que determina **como** fazer aquele processo. (ver **w_salesrep2.w**)



Como o ADM funciona (cont.)

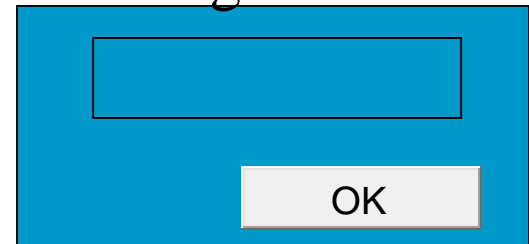
- A diferença entre um código na versão 7 e para o Progress 8.

Progress 7.3



On choose of bt-ok do:
assign field1 field2.
end.

Progress 8

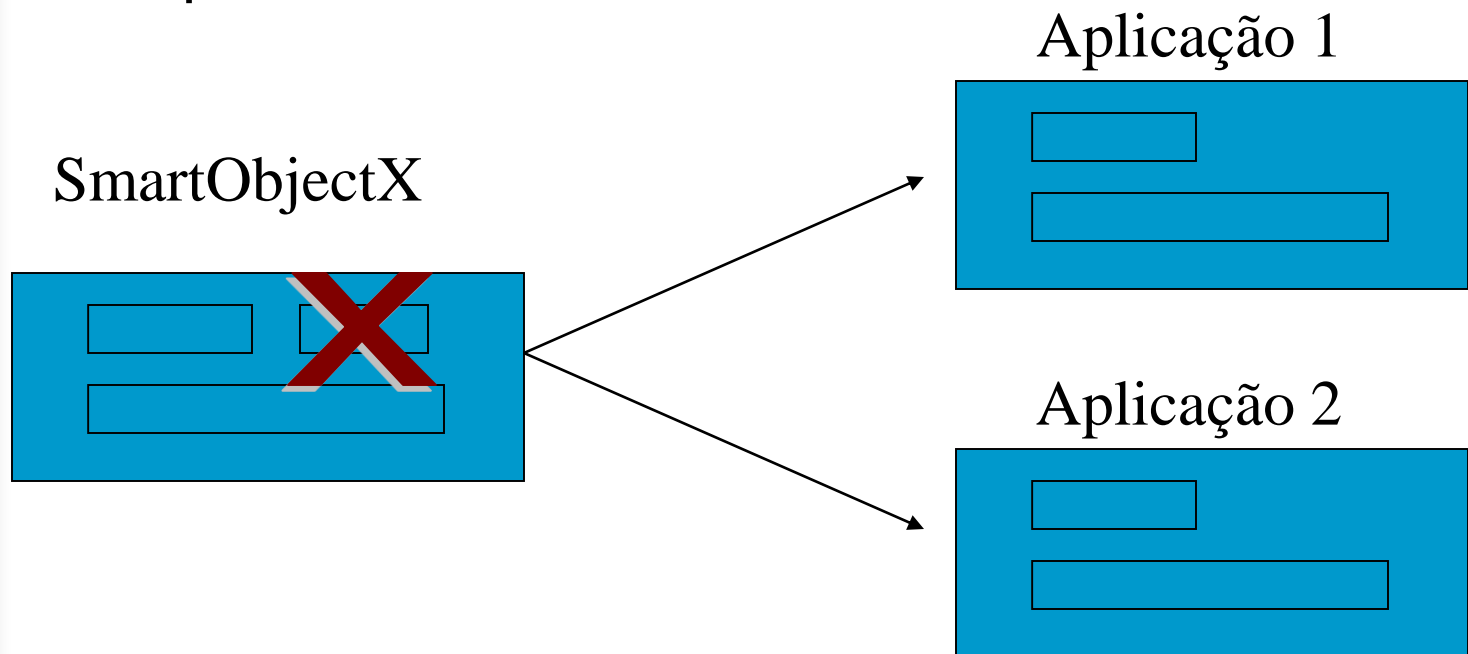


On choose of bt-ok do:
run notify ("update-record").
end.

Procedure adm-update-record;

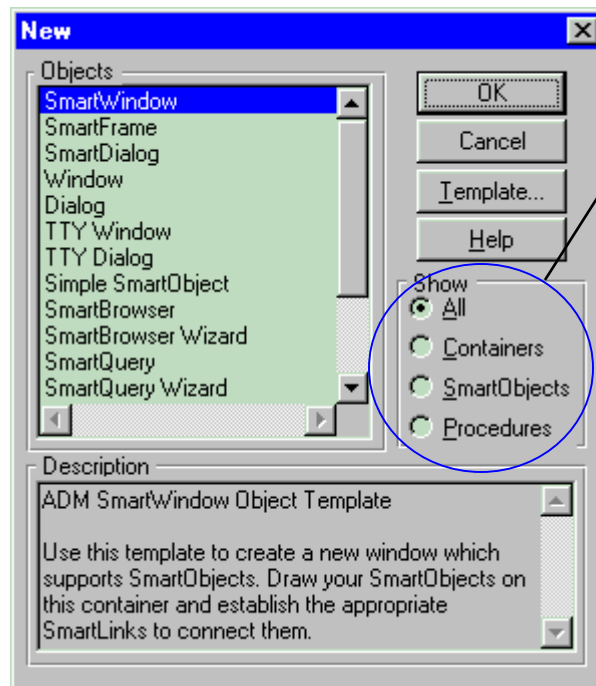
Como o ADM funciona (cont.)

- Modificando-se um SmartObject, a diferença será replicada.



Suporte UIB para SmartObjects

- Criando-se aplicações com SmartObjects, o UIB gerará código tornando-o parte do .w para o SmartContainer.
 - Ao colocar um SmartObject dentro de um SmartContainer, o UIB escreverá o código para inicializar na tela cada SmartObject deste container.
 - Utilizando-se o Link Editor, o UIB escreverá o código dentro do SmartContainer para definir links entre os SmartObjects.

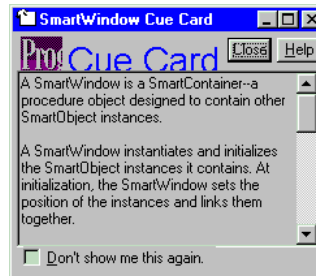


O Tela **NEW** possibilitará a escolha por SmartObjects, podendo esta escolha ser por :

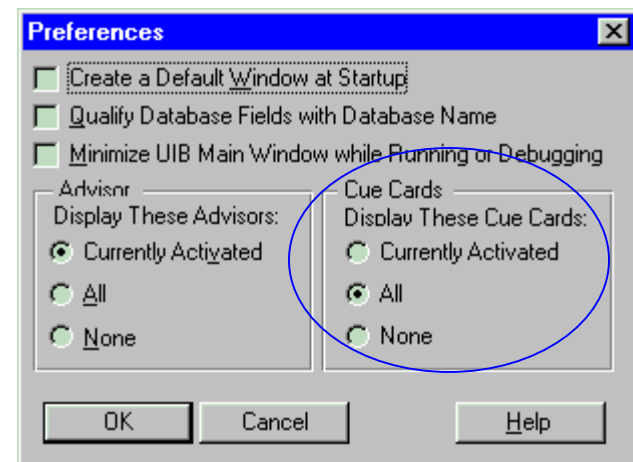
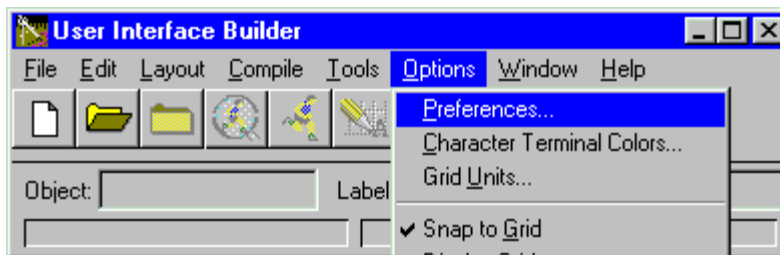
- Containers,
- SmartObjects,
- Procedures ou ALL.

Suporte UIB para SmartObjects (cont.)

- Nota-se a existência de CueCard para ajudar no desenvolvimento.



- Para eliminarmos os CueCards basta selecionar **Options / Preferences** e em CueCards selecionar **NONE**. Tais preferências serão salvas no arquivo .ini.





Como construir aplicações utilizando SmartObjects

- Selecione NEW.
- Selecione o container.
- Selecione o SmartObject através da Paleta, o SmartObject será selecionado.
- Click no Container que você quer colocar o SmartObject.
- Aceite as sugestões de links ou utilize o Link Editor.
- Execute, para fechar ou pressione Stop ou feche o Container.

Laboratório

- Desenvolver uma aplicação para consulta dos Representantes.

Utilizar :

- SmartWindow,
- SmartViewer,
- SmartQuery,
- SmartPanel p-Navico.r,
- Botão DONE.



- Salvar como w-<seu-nome>1.w

Laboratório (cont.)

- Desenvolver uma aplicação para consulta **Customer** dos **Salesrep.**

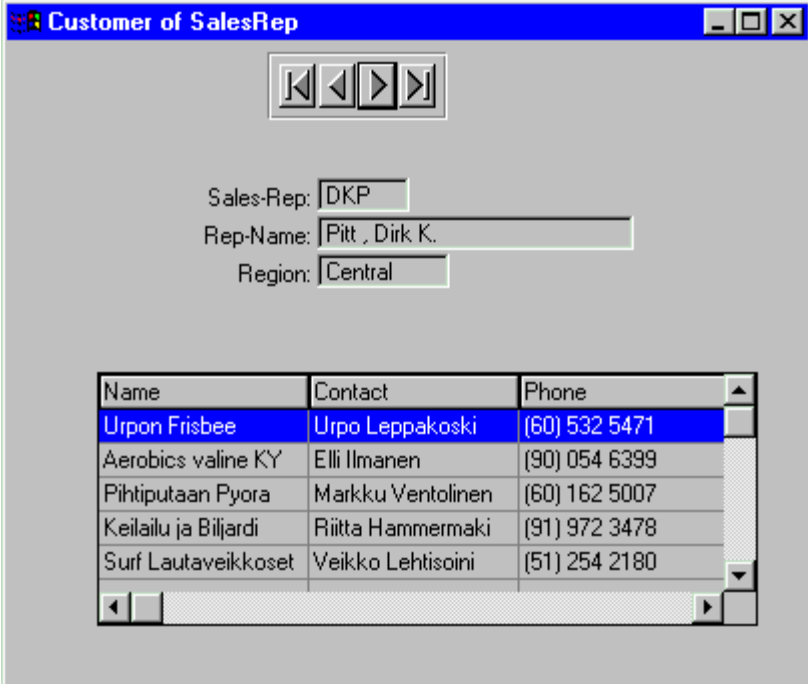
Utilizar :

- SmartWindow,
- SmartViewer para SalesRep,
- SmartQuery para SalesRep,
- SmartPanel Navico.r,
- **SmartBrowser para Customer utilizando o SmartBrowser Wizard**

(**Atenção! para a External table informar a tabela Salesrep**),

- Botão DONE.

- Salvar como w-<seu-nome>2.w



Name	Contact	Phone
Urpon Frisbee	Urpo Leppakoski	(60) 532 5471
Aerobics valine KY	Elli Ilmanen	(90) 054 6399
Pihtiputaan Pyora	Markku Ventolinen	(60) 162 5007
Keilailu ja Biljardi	Riitta Hammermaki	(91) 972 3478
Surf Lautaveikkoset	Veikko Lehtisoini	(51) 254 2180

Laboratório (cont.)

- Desenvolver aplicação para a tabela Customer
Utilizar:
 - SmartBrowser,
 - SmartViewer,
- Não aceitar o Link do Progress Advisor.
- Criar o Link de **Record** entre SmartBrowser e a SmartViewer.
- Executar. (ver w-jaison3.w)

The screenshot shows a window titled "Customer Address Information". It contains a table with 3 columns: "Cust-Num", "Name", and "Contact". The table has 6 rows of data. The 4th row, "Go Fishing Ltd" with contact "Alan Frogbrook", is highlighted with a blue dashed border. Below the table is a form with labels and text boxes for "Name", "Address", "Address2", "Country", "City", and "State". The values entered are "Go Fishing Ltd", "Unit 2", "83 Ponders End Rd", "United Kingdom", "Harrow", and "Middlesex". A "Done" button is at the bottom right.

Cust-Num	Name	Contact
1	Lift Line Skiing	Gloria Shepley
2	Urpon Frisbee	Urpo Leppakoski
3	Hoops Croquet Co.	Michael Traitser
4	Go Fishing Ltd	Alan Frogbrook
5	Match Point Tennis	Robert Dorr
6	Fanatical Athletes	Andrew Strong

Name: Go Fishing Ltd
Address: Unit 2
Address2: 83 Ponders End Rd
Country: United Kingdom
City: Harrow
State: Middlesex

Done

Laboratório (cont.)

- Desenvolver uma aplicação para a Tabela Customer com navegação e display de dados. (ver w4.w)

Utilizar:

- Cust-num, name, contact e phone number.

Quais objetos serão necessários ?

SmartWindow,
SmartPanel(p-navico.r).
SmartViewer.
SmartQuery.



Laboratório (cont.)

- Desenvolver aplicação que mostre os Pedidos do Cliente (ver w5.w).

Utilizar:

- SmartWindow,
- SmartPanel(p-navico.r),
- SmartQuery(para a tabela Customer).
- SmartViewer(with auto field).
- SmartBrowser Wizard(Order of Customer).



Laboratório(cont.)

- Desenvolver aplicação conforme a figura.
(Ver w6.w)

The screenshot shows a Windows application window titled 'w6'. At the top, there are four navigation buttons: back, forward, and two others. Below these are input fields for 'Cust-Num: 1' and 'Name: Lift Line Skiing'. The main area contains two tables. The first table, 'Order of Customer', has columns for Order-num, Ordered, Instructions, and Promised. The second table, 'Order-line of Order', has columns for Item-num, Qty, Price, Discount, and Item-Name. Both tables have a vertical scrollbar on the right.

Order-num	Ordered	Instructions	Promised
6	05/01/9		03/02/93
36	19/01/9		04/03/93
79	10/02/9		26/02/93
177	30/03/9		09/04/93
185	05/04/9		19/05/93

Item-num	Qty	Price	Discount	Item-Name
00009	67	18,00	35%	Swim Goggles
00009	12	18,00	35%	Swim Goggles
00011	34	34,00	35%	Lacrosse Stick
00019	20	2,75	35%	Ski Wax - Red

Observação do laboratório anterior

- Observar os Links.

w6 - w6.w

Navegação

Record

Cust-Num:

Name:

q6.w

Order of Customer

Order-num	Ordered	Instructions	Promised

Order-line of Order

Item-num	Qty	Price	Discount	Item-Name

Record

Links		
Source	Link Type	Target
h_b6	Record	h_b6a
h_p-navico	Navigation	h_q6
h_q6	Record	h_v6
h_q6	Record	h_b6

Laboratório(cont.)

- Construir uma aplicação conforme abaixo.
(Ver w7.w)

	Order-num	Ordered	Instructions	Promised
	39	20/01/93		18/02/93
	71	05/02/93		17/02/93
	142	11/03/93		16/04/93
	175	29/03/93		11/05/93

Navegation

Record

Record

Tableio

Laboratório(cont.)

- Desenvolver a aplicação abaixo (ver w8.w).

The application window displays the following data:

Cust-Num	Name
1	Lift Line Skiing
2	Urpon Frisbee
3	Hoops Croquet Co.
4	Go Fishing Ltd
5	Match Point Tennis
6	Fanatical Athletes

Order-num	Ordered	Instructions	Promised
6	05/01/93		03/02/93
36	19/01/93		04/03/93
79	10/02/93		26/02/93
177	30/03/93		09/04/93
185	05/04/93		19/05/93

Form fields:

Ordered: 05/01/93
Instructions:
PO:
Promised: 03/02/93
Shipped: 08/02/93
Terms: Net30
Carrier: Standard Mail

Buttons: Save, Reset, Add, Copy, Delete, Cancel

Laboratório(cont.)

Local-initialize (smart-window)

run pi-recebe-viewer in h_cust (input h_viewer)

Value-changed of b_cust

run pi-recebe-cust in h-viewer (input cust-num).

W8

Cust-Num Name

1	Lift Line Skiing
2	Urpon Fishbee
3	Hoons Croquet Co.
4	Golfishing Ltd
5	Match Point Tennis
6	Manatical Athletes

Order of Customer

Order-num	Ordered	Instructions	Promised
6	05/01/93		03/02/93
36	19/01/93		04/03/93
79	10/02/93		26/02/93
177	30/03/93		09/04/93
185	05/04/93		19/05/93

Ordered: 05/01/93

Instructions:

PO:

Promised: 03/02/93

Shipped: 08/02/93

Terms: Net30

Carrier: Standard Mail

Save Reset Add Copy Delete Cancel

Local-add-record

assign order.cust-num:screen-value = string(i-cust-num).

Local-assign-record

assign order.cust-num = i-cust-num.

Criação de SmartObjects

Esta sessão visa melhorar o desenvolvimento e a visão na criação de SmartObjects.

■ Convenção de nomes:

SmartWindow	w-nome
SmartDialog	d-nome
SmartFrame	f-nome
SmartQuery	q-name
SmartBrowser	b-name
SmartPanel	p-name



SmartViewer

- Existem dois tipos de SmartViewer:
 - SmartViewer, necessita adição dos Dbfields,
 - SmartViewer with Auto-Field, a adição dos Dbfields é automática.



SmartQuery

- Também existe dois tipos de criação:
 - utilizando o Wizard ou sem a ajuda deste.
- Utilize sempre o Wizard pois a implementação é mais fácil e não muito demorada.

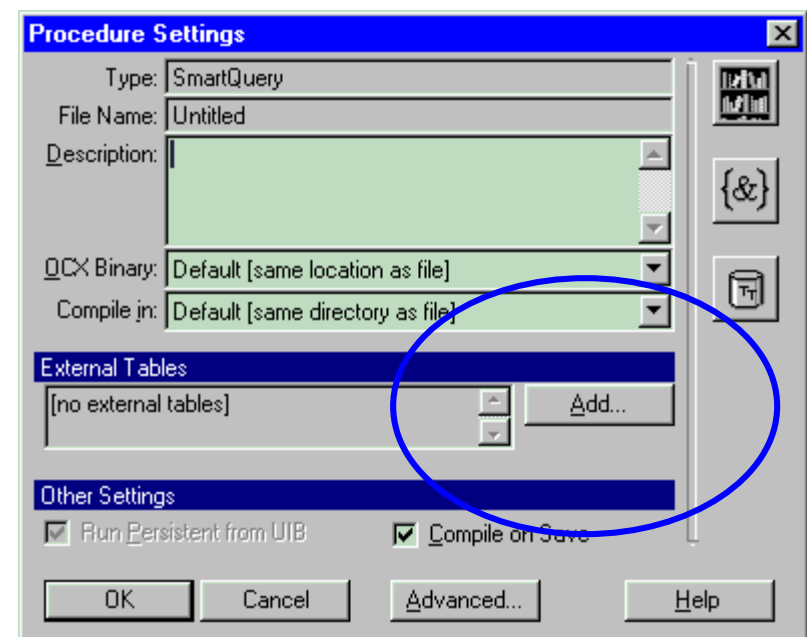
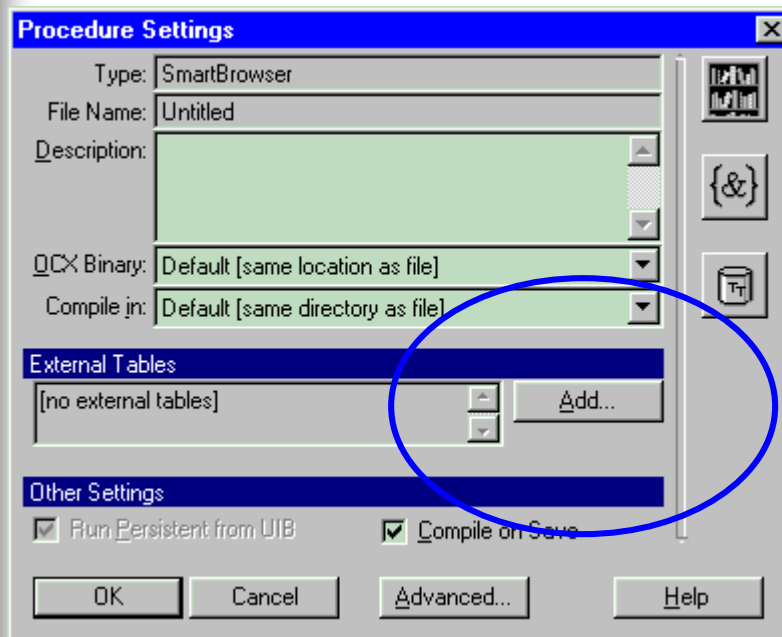


SmartBrowser

- Mais um SmartObject que possui Wizard.
 - Pode-se utilizar o Wizard pela facilidade no desenvolvimento.

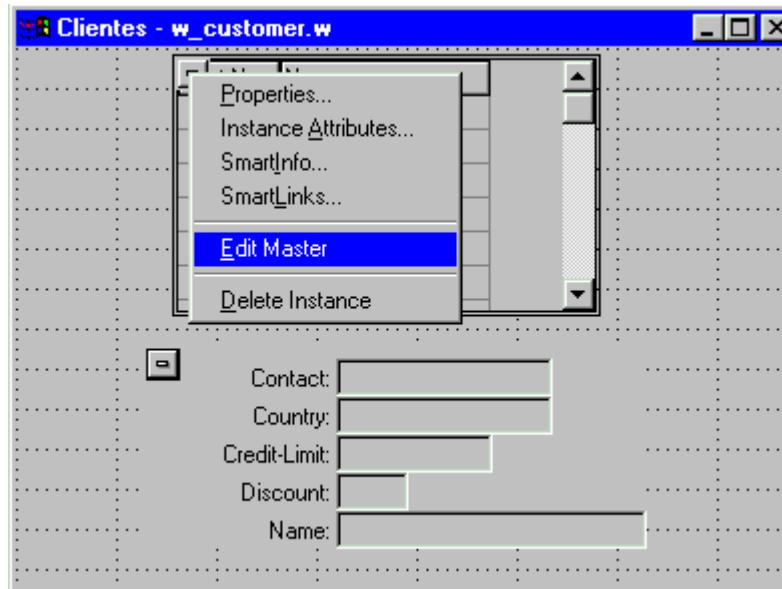
External Tables

- Para SmartBrowser e SmartQuery(sem wizard) adiciona-se External Tables através do menu Tools / Procedure Settings do UIB.



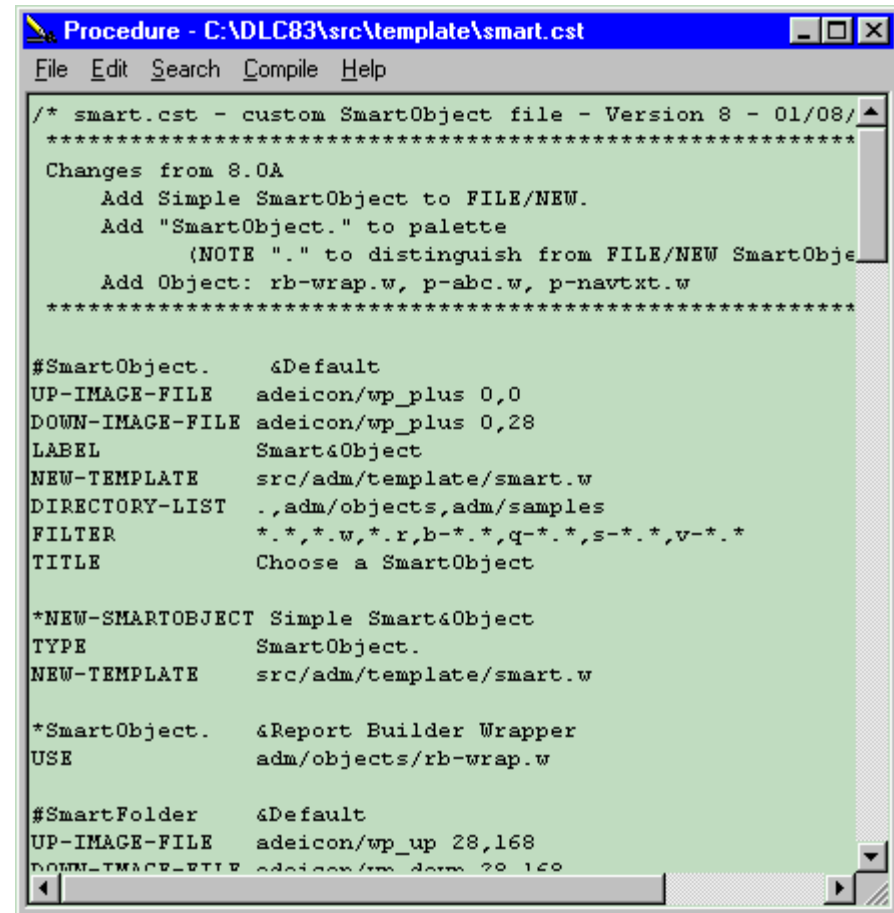
Como editar um SmartObject

- Clicar no sinal de 'menos' do SmartObject e selecionar **EditMaster**. Altere e salve o SmartObject.



Smart.cst

- É um arquivo texto externo, que utiliza uma sintaxe própria, possuindo as definições de objetos do UIB. Encontra-se em **src\template**.



The screenshot shows a text editor window titled "Procedure - C:\DLC83\src\template\smart.cst". The window contains the following text:

```
/* smart.cst - custom SmartObject file - Version 8 - 01/08/
*****
Changes from 8.0A
    Add Simple SmartObject to FILE/NEW.
    Add "SmartObject." to palette
        (NOTE "." to distinguish from FILE/NEW SmartObject)
    Add Object: rb-wrap.w, p-abc.w, p-navtxt.w
*****

#SmartObject.      &Default
UP-IMAGE-FILE      adeicon/wp_plus 0,0
DOWN-IMAGE-FILE    adeicon/wp_plus 0,28
LABEL              Smart&Object
NEW-TEMPLATE        src/adm/template/smart.w
DIRECTORY-LIST      .,adm/objects,adm/samples
FILTER              *.*,*.w,*.r,b-*,q-*,s-*,v-*.
TITLE              Choose a SmartObject

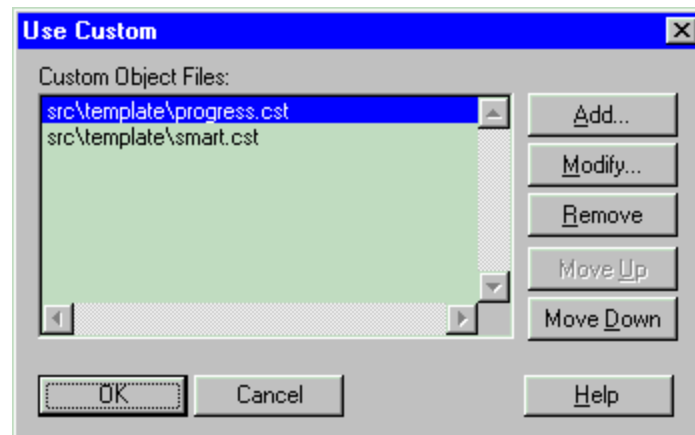
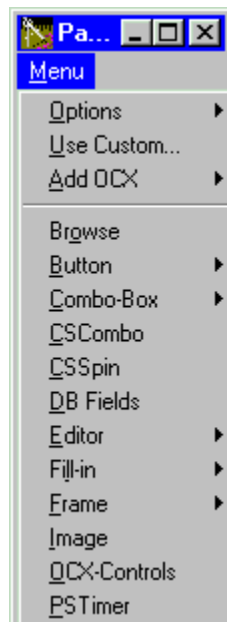
*NEW-SMARTOBJECT Simple Smart&Object
TYPE                SmartObject.
NEW-TEMPLATE        src/adm/template/smart.w

*SmartObject.      &Report Builder Wrapper
USE                 adm/objects/rb-wrap.w

#SmartFolder        &Default
UP-IMAGE-FILE        adeicon/wp_up 28,168
DOWN-IMAGE-FILE      adeicon/wp_down 28,168
```

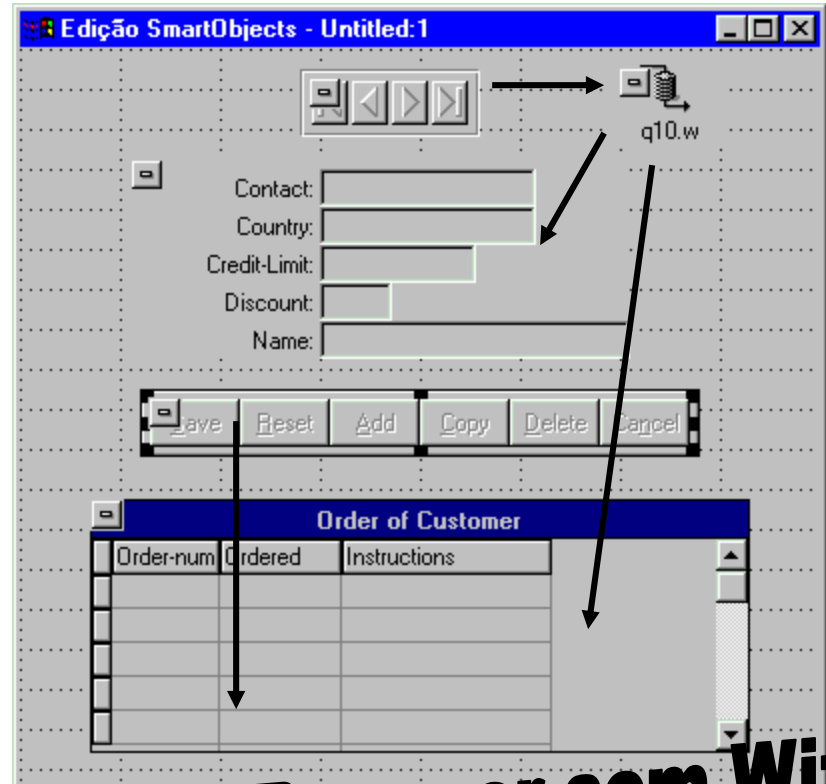
Smart.cst (cont.)

- Através do menu / Use Custom... na paleta de objetos, pode-se redefinir o arquivo .cst a ser utilizado.



Laboratório

- Desenvolver aplicação conforme a imagem.
- Executar.
- Editar o SmartBrowser, habilitando-se o campo **Utilizar o SmartBrowser sem Wizard** *order.instructions* . (ver w10.w)



Utilizar o SmartBrowser sem Wizard



Links

- É uma conexão entre dois SmartObjects que define como estes interagirão com qualquer outro e o tipo de mensagens que eles poderão passar.



Links

- **Navigation:** Passa a requisição para ir para o primeiro, último, anterior ou posterior.
- **Record:** Passa o Rowid do registro.
- **Tableio:** Requisita a adição, eliminação, atualização de um registro específico.



Links (cont.)

■ **Container:** Link para cada SmartObject inserido ao container.

■ **Page:** Link do SmartFolder e outras páginas com este Container.

■ **Page:** Link de todos os SmartObjects com uma página específica.

*Estes links são automaticamente criados pelo Progress.

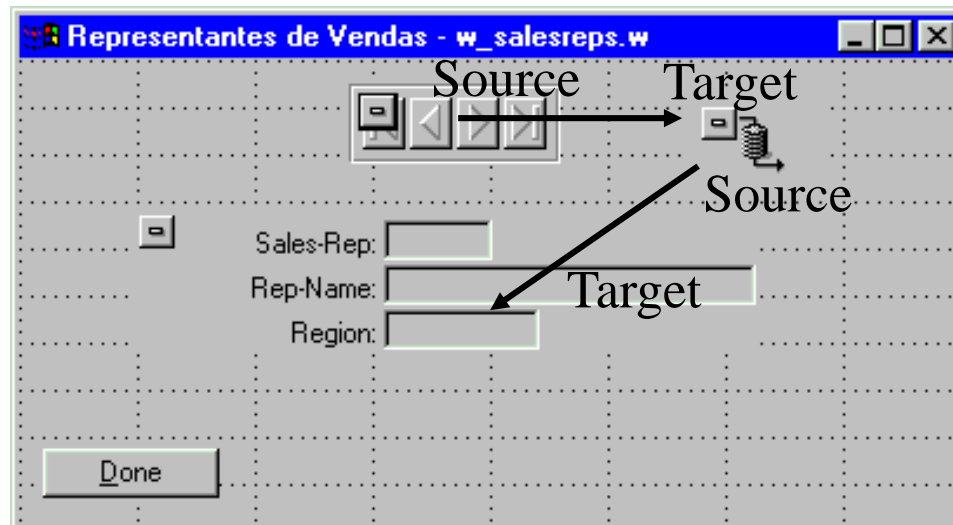


Links (cont.)

- **State:** É um caminho que passa informação entre dois SmartObjects que não estão conectados, através de outro tipo de link.
- **Group-assign:** Combina vários SmartObjects para que cada UPDATE de um dependa do UPDATE dos outros.
- **User-defined:** Pode-se definir o próprio tipo de link com adição de uma mensagem.

Introdução aos Links

- Cada link conecta sempre dois SmartObjects.
- Você *sempre* definirá um SmartObjects como *Source(fonte)* e um outro SmartObjects como *Target(alvo)*.





Procedures Persistent

- São programas como qualquer outro, mas que ficarão em memória ate serem eliminados.
- Necessidades:
 - variável handle(sera o ponteiro da procedure).
 - Executar o programa e carregar o valor da variável handle(run mat.p set h)
 - Deletar o programa no final da execução.



Procedures Persistents(cont.)

- Todo smartobject é persistent.
- Pode-se verificar através do código fonte do container a existência de h_nome-físico.
Ex. define variable h_b-customer as handle no-undo.



Mensagens

- E o processo pelo qual os smartobjects se comunicam.
- Cada smartobject pode enviar e receber mensagens.
- Os tipos de mensagens que um smartobject poderá receber ou mandar será determinado pelo tipo de link.



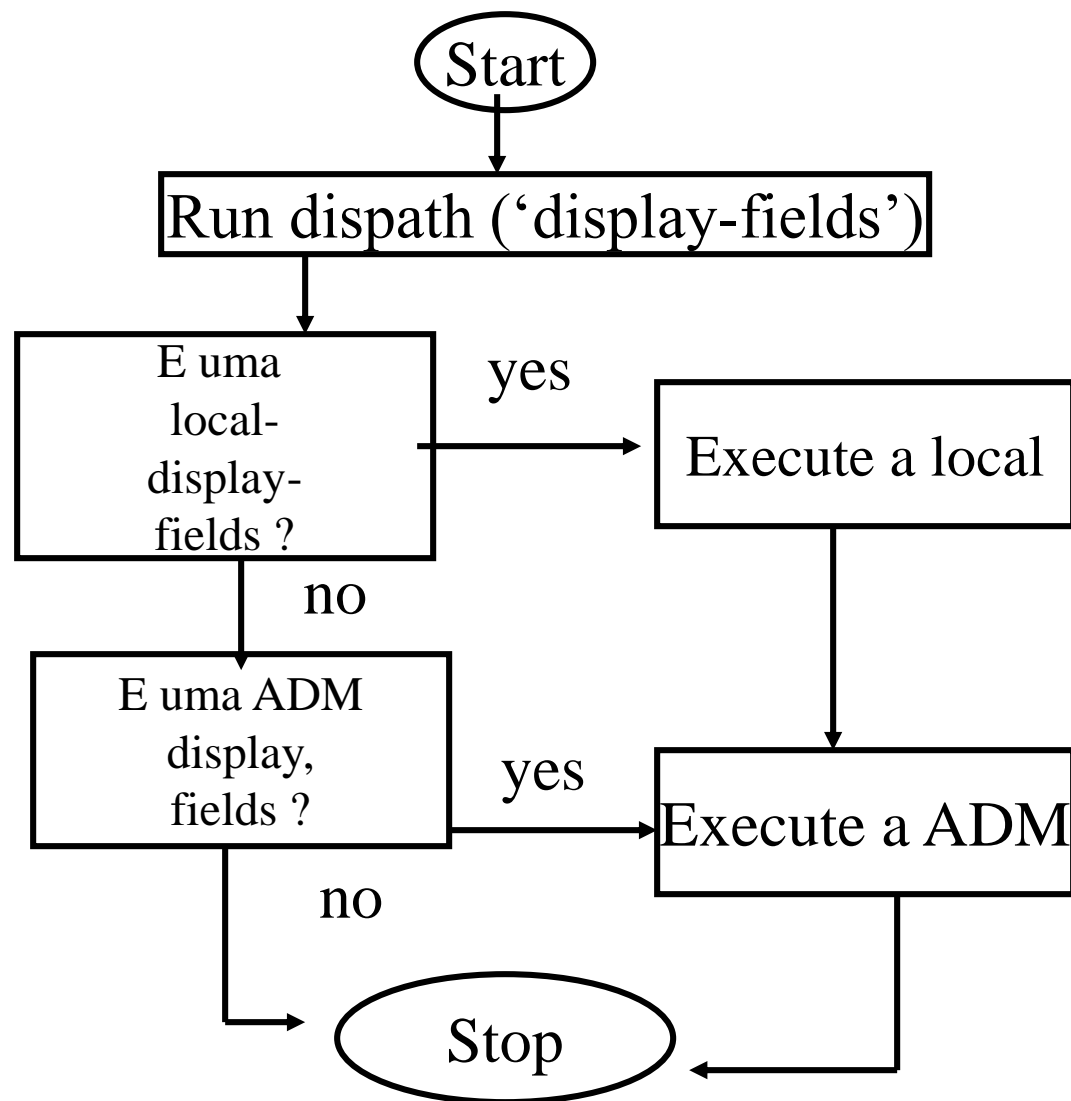
Mensagens(cont.)

- O código ADM inclui dois tipos de procedures.
 - Event: procedure que pode ser customizada.
 - Method: procedure que não pode ser customizada somente utilizada.

Mensagens(cont.)

- Para se customizar Event deve-se criar uma nova procedure local.
Ex.
Local-open-query.
Local-assign-records.
- Sintaxe
run dispatch [in handle] ('base-procedure').
Ex. run dispatch in h_viewer ('display-fields').

Mensagens(cont.)





Mensagens(cont.)

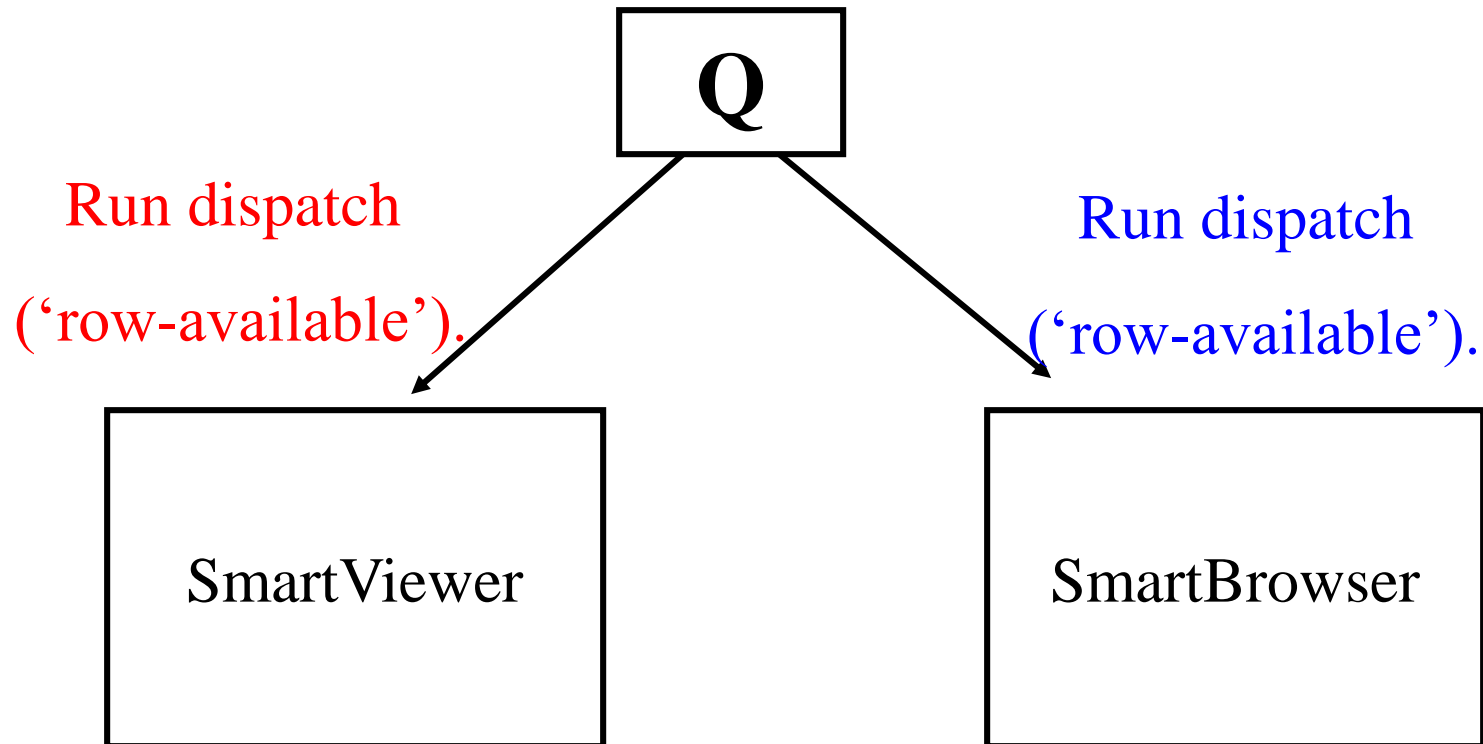
Notify: Tem como propósito executar o run dispatch para Vários objetos. Identificando todos os alvos de um fonte (lendo uma tabela de links), então ele executará um dispatch para cada objeto alvo.

Sintaxe:

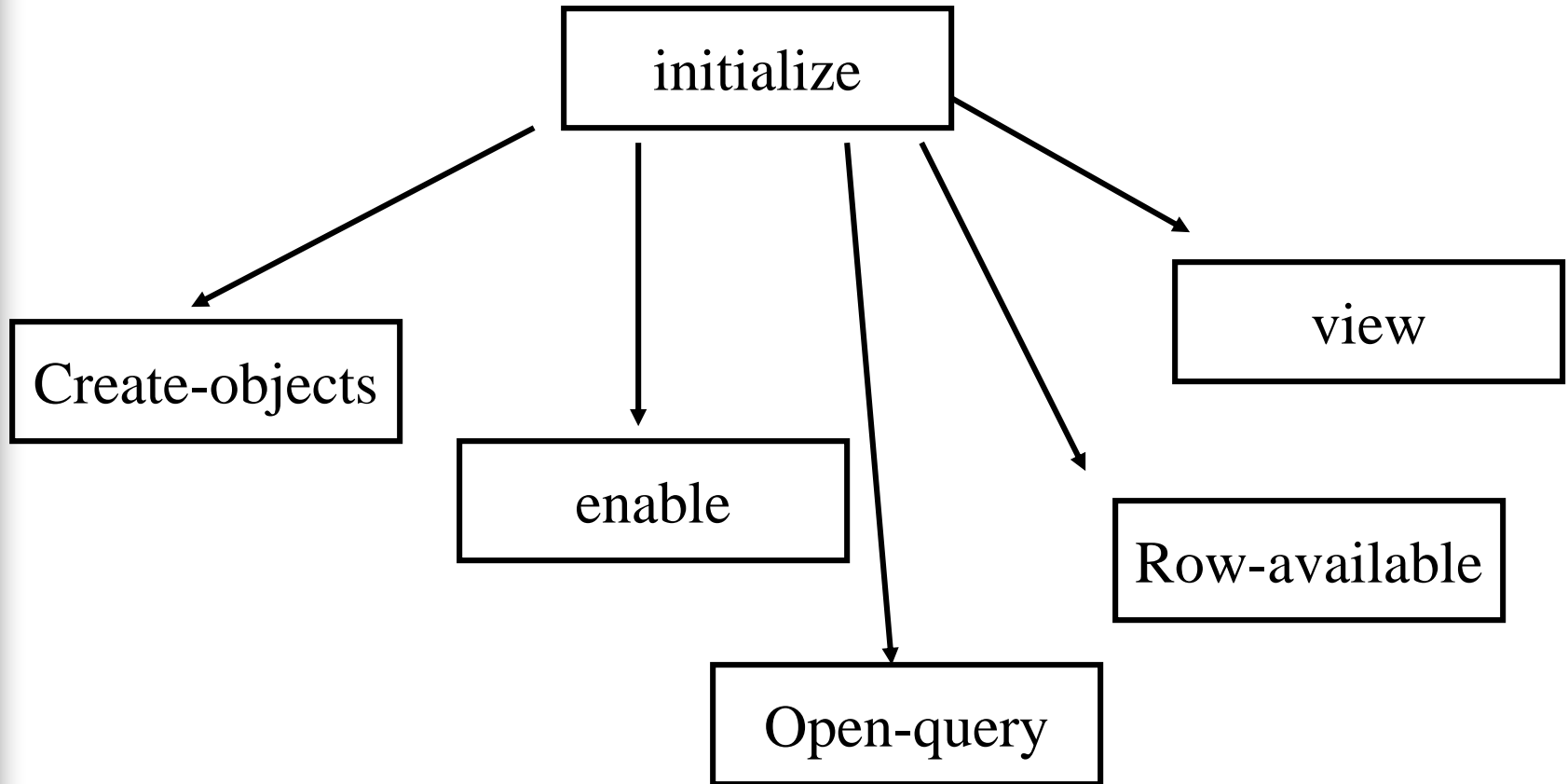
```
run notify [in handle] ('base-procedure',[link-  
types]).
```

Mensagens (cont.)

Ex. run notify in this-procedure ('row-available').

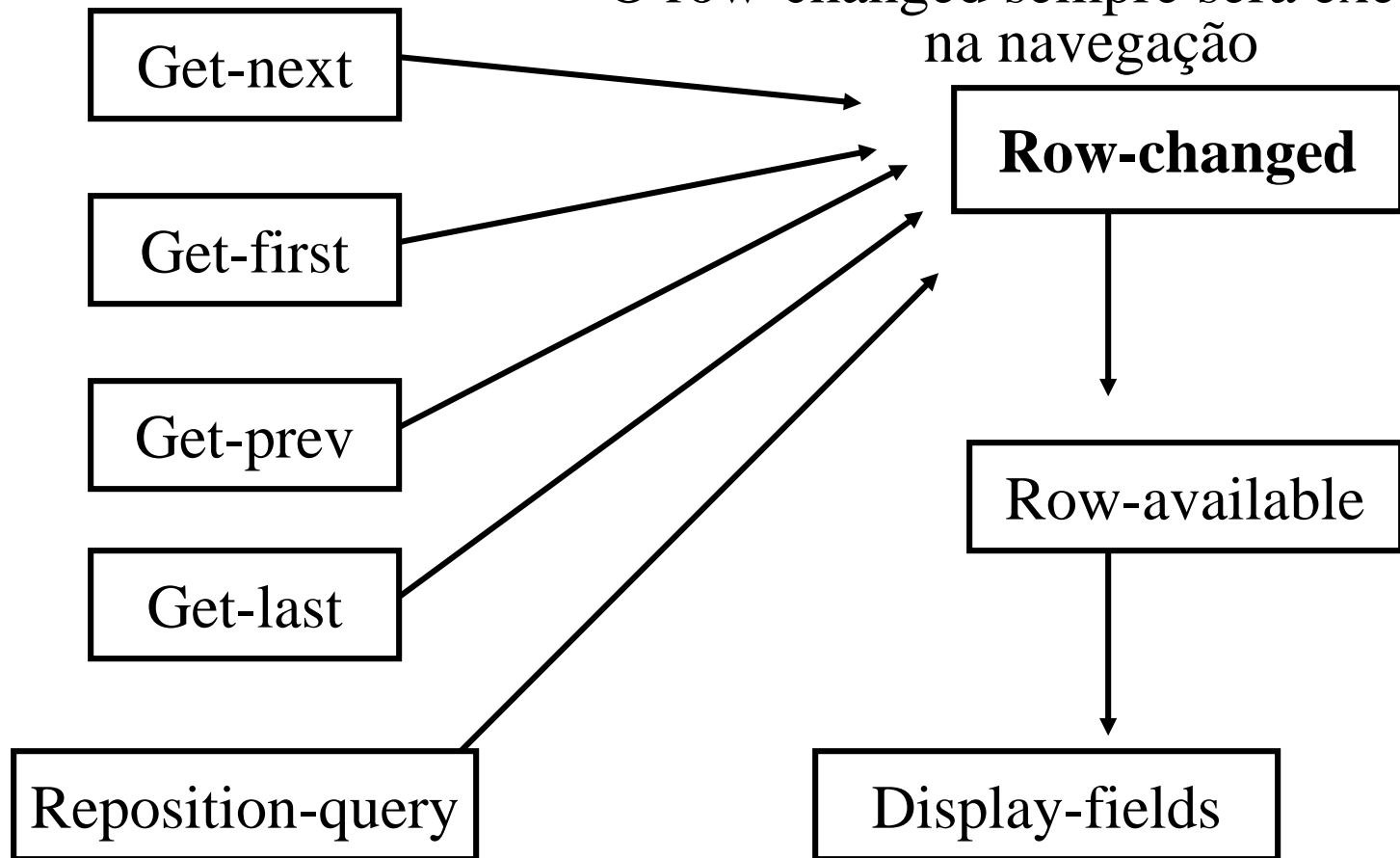


Mapas de eventos

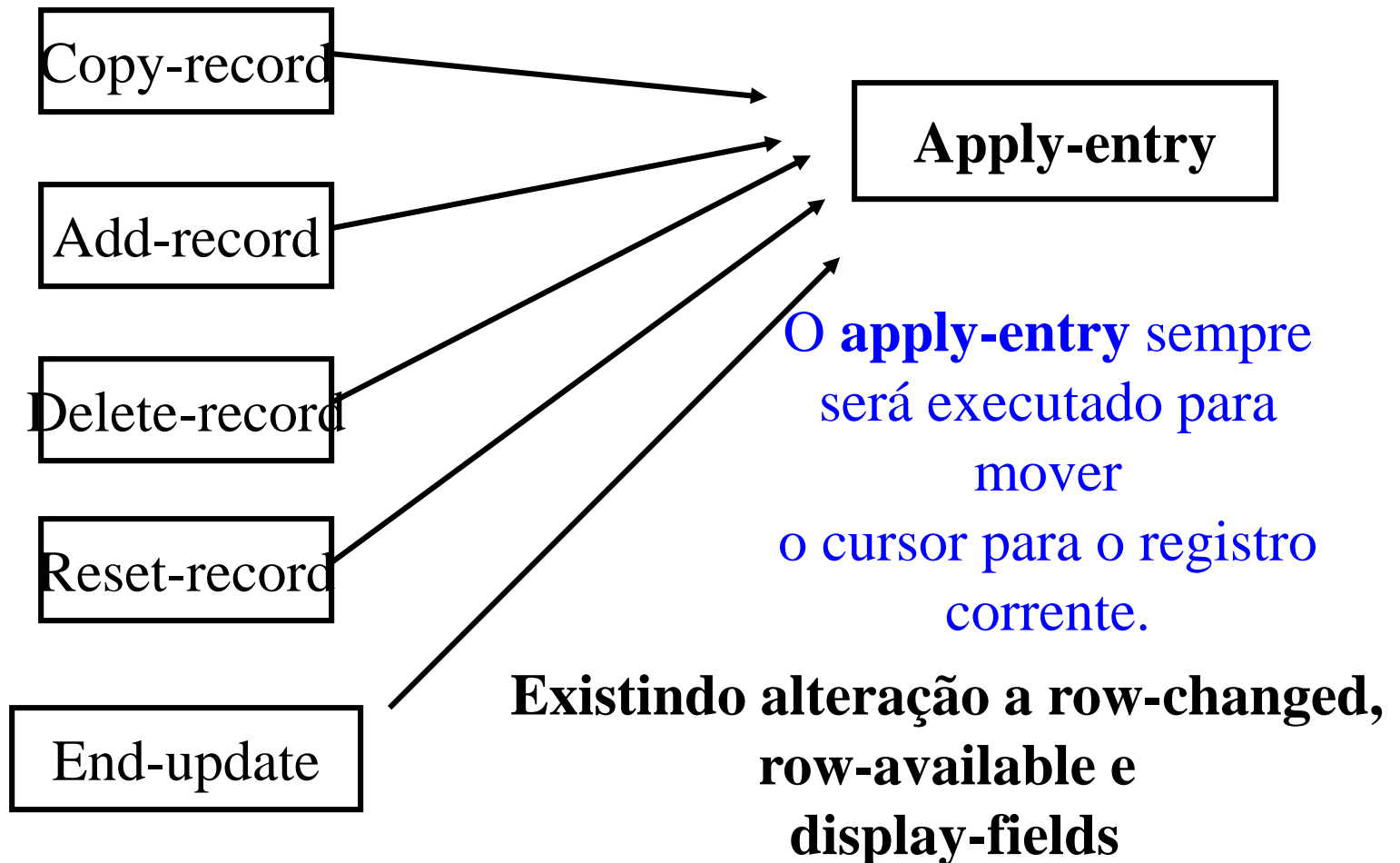


Mapas de eventos(cont.)

O row-changed sempre será executado na navegação



Mapas de eventos(cont.)



Mapas de eventos(cont.)

