



# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner



**DEV-40: Using SmartDataObjects (SDO) with the Advanced GUI**

Mike Fechner, Consultingwerk Ltd.  
Progress Exchange 2008, Paris, France



---

---

---

---

---

---

---



**Consultingwerk Ltd.**

- Independent Consulting Organisation
- Located in Cologne, Germany
- Customers in Germany, Europe and USA
- Specialized on OpenEdge platform
  - Dynamics®
  - ADM2
  - OERA
  - .NET™, OpenClient
  - Advanced GUI for OpenEdge® 10.2A

DEV-40: Using SmartDataObjects with the Advanced GUI 2 

---

---

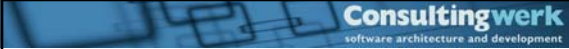
---

---

---


---

---



**Agenda**

- Introduction
- SmartDataObjects Reviewed
- Data Binding in the Advanced GUI
- Basic Data Binding with SmartDataObjects
- Controlling Batching and Sorting
- Updating
- Summary

DEV-40: Using SmartDataObjects with the Advanced GUI 3 

---

---

---

---


---

---

---


# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

  
software architecture and development

### Introduction

- Consultingwerk has developed a replacement for the Dynamics rendering engine
- Dynamics4.NET is based on the Advanced GUI for OpenEdge 10.2A and Dynamics and offers "load and go" migration to the .NET user interface
- Compatibility between .NET UI and ADM2 major design goal to support migration of business logic as well as repository information

DEV-40: Using SmartDataObjects with the Advanced GUI 4 

---

---


---

---

---


---

---

  
software architecture and development

### Agenda

- Introduction
- SmartDataObjects Reviewed
- Data Binding in the Advanced GUI
- Basic Data Binding with SmartDataObjects
- Controlling Batching and Sorting
- Updating
- Summary

DEV-40: Using SmartDataObjects with the Advanced GUI 5 

---

---


---

---

---


---

---

  
software architecture and development

### SmartDataObjects Reviewed

- Important part of ADM2 Architecture
- Business Logic component, data retrieval and update
- Introduced in V9.0, continuously enhanced in V9.1, OpenEdge 10
- Built using
  - persistent procedures, super procedures
  - dynamic queries
  - temp-tables

DEV-40: Using SmartDataObjects with the Advanced GUI 6 

---

---

---

---

---

---

---

# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

**Consultingwerk**  
software architecture and development

### SmartDataObjects Reviewed

- Usage in ADM2 applications driven by wizards (AppBuilder supported)
- Usage outside ADM2 applications possible and documented ("Open SDO" white paper from John Sadd available on PSDN)

DEV-40: Using SmartDataObjects with the Advanced GUI 7

---

---

---

---

---

---

---

---

**Consultingwerk**  
software architecture and development

### Agenda

- Introduction
- SmartDataObjects Reviewed
- **Data Binding in the Advanced GUI**
- Basic Data Binding with SmartDataObjects
- Controlling Batching and Sorting
- Updating
- Summary

DEV-40: Using SmartDataObjects with the Advanced GUI 8

---

---

---

---

---

---

---

---

**Consultingwerk**  
software architecture and development

### Data Binding in the Advanced GUI

- **Progress.Data.BindingSource**
  - Provides data for .NET Controls
    - Implements required APIs for .NET Controls (IList)
    - OpenEdge data provided as .NET needs
    - Any ABL Query ( TT, TABLE ), ProDataSet™

DEV-40: Using SmartDataObjects with the Advanced GUI 9

---

---

---

---

---


---

---

---


# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

  
software architecture and development

### Data Binding in the Advanced GUI

- ABL behavior
  - Automatic data synchronization
  - Automatic sorting
  - Automatic updating
  - Automatic batching
  - Automatic currency
- Properties
  - AllowEdit, AllowNew, AllowRemove
  - NewRow, RowModified
  - Position, Count
  - InputValue, ChildInputValue

DEV-40: Using SmartDataObjects with the Advanced GUI 10 

---

---

---


---

---

---

---

---

  
software architecture and development


### Data Binding in the Advanced GUI

- Simple .NET control ( UltraEdit )  

```
editBox:DataBindings.Add ( "Text", pbs, "OrderNum" );
```
- .NET table-like control ( UltraGrid )  

```
grid:DataSource = pbs.
```
- .NET list control ( UltraListView )  

```
list:DataSource = pbs.  
list:DataTextField = "State".  
list:DataValueField = "State-Name".
```

DEV-40: Using SmartDataObjects with the Advanced GUI 11 

---

---

---


---

---

---


---

---

  
software architecture and development

### Agenda

- Introduction
- SmartDataObjects reviewed
- Data Binding in the Advanced GUI
- **Basic Data Binding with SmartDataObjects**
- Controlling Batching and Sorting
- Updating
- Summary

DEV-40: Using SmartDataObjects with the Advanced GUI 12 

---

---

---

---

---


---

---

---

# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner




software architecture and development

### Data Binding with SmartDataObjects

- Use similar concepts as SmartDataBrowser
- Bind .NET Controls to ProBindingSource (Progress.Data.BindingSource) control
- Get the **DataHandle** of the SDO
  - the query to client side RowObject temp-table  
`{get DataHandle hDataHandle hSDO}`
- Open the SDO Query (if not done at initialization)
- Assign the **HANDLE** Property of the ProBindingSource control

DEV-40: Using SmartDataObjects with the Advanced GUI

13



---

---

---


---

---

---

---

---



software architecture and development

### Sample without Batching


```
METHOD PRIVATE VOID InitializeDataObject ( ) :  
    RUN samples/SDOForm/dcustomer.w PERSISTENT SET hSDO .  
  
    {set OpenOnInit True hSDO} .  
    {set RowsToBatch 0 hSDO} .  
  
    RUN initializeObject IN hSDO .  
  
    {get DataHandle hDataQuery hSDO} .  
  
    bindingSource1:HANDLE = hDataQuery .  
  
    bindingSource1:Batching = FALSE .  
END.
```

Launch SDO as persistent procedure

Get handle to RowObject query and assign to Binding Source

DEV-40: Using SmartDataObjects with the Advanced GUI

14



---

---

---


---

---

---

---

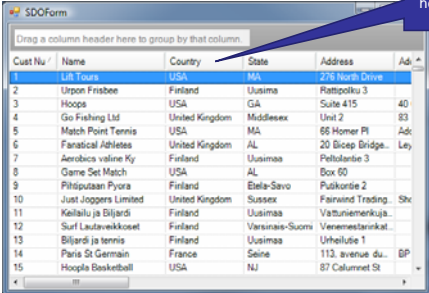
---



software architecture and development


### Sample without Batching

Drag Country and State column headers to group by area...



DEV-40: Using SmartDataObjects with the Advanced GUI

15



---

---

---

---

---

---

---

---

# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

**Consultingwerk**  
software architecture and development

### Sample without Batching

... to group SDO resultset by Country and State

Country: / State: /

- Country: Australia (1 item)
- Country: Austria (2 items)
- Country: Finland (11 items)
- Country: France (5 items)
- State: (2 items)
- State: Bretagne (2 items)
- State: Midi (1 item)
- State: Rhone (1 item)
- State: Seine (1 item)
- Country: Italia (1 item)
- Country: Nederland (1 item)

Cust Nu	Name	Address	Address2	City
36	La voile en folie	rue de l'Océan	allée de Paris	Brest
77	Le Hollandais volant	35, rue des Oise...	3ème étage	Rennes

DEV-40: Using SmartDataObjects with the Advanced GUI 16

---

---

---

---

---

---

---

---

**Consultingwerk**  
software architecture and development

### Designing the Form

- Assigning the **HANDLE** property of the ProBindingSource creates schema information during run time
- To „design“ the fields in the UltraGrid the schema information is required at design time
- The ProBindingSource designer can import an XML schema definition (XSD)
- XSD can be generated from temp-table
- RowObject** temp-table contains custom fields (SDO include file) and ADM2 control fields

DEV-40: Using SmartDataObjects with the Advanced GUI 17

---

---

---

---

---

---

---

---

**Consultingwerk**  
software architecture and development

### Generating XML Schema Definition

```
<SCOPED-DEFINE DATA-FIELD-DEFS (samples/SDOForm/dcustomer.1)
  DEFINE TEMP-TABLE RowObject NO-UNDO
  {<DATA-FIELD-DEFS>
    {sdo/adm2/objfields.1}
    *
  }
  /* ***** Main Block ***** */
  TEMP-TABLE RowObject:WRITE-XMLSCHEMA ("FILE", "samples/SDOForm/dcustomer.xsd",
    TRUE, /* formatted */
    ?, /* Default encoding */
    FALSE /* no min schema */).
```

DEV-40: Using SmartDataObjects with the Advanced GUI 18

---

---

---

---

---

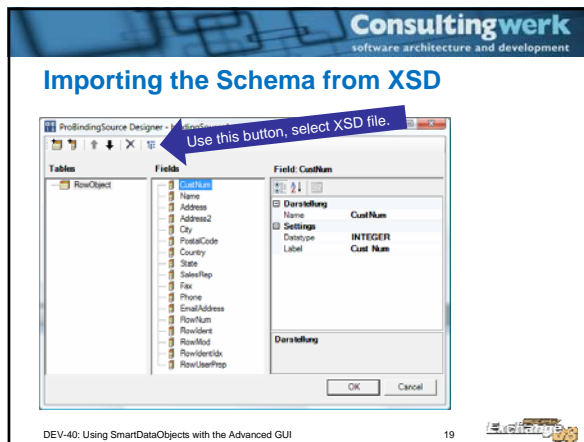
---

---

---

# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner



---

---

---

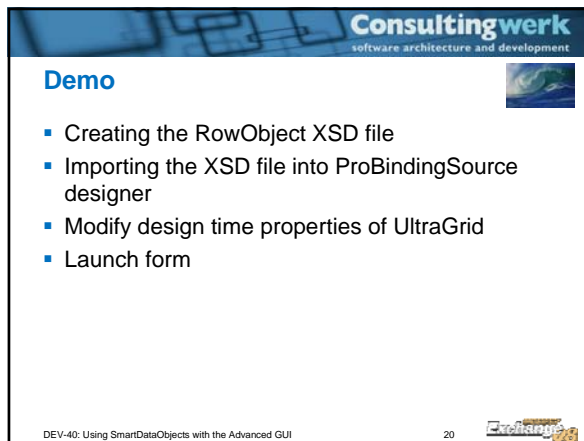
---

---

---

---

---



---

---

---

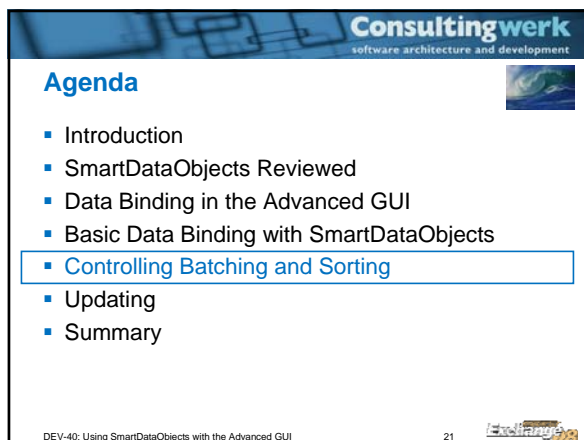
---

---

---

---

---



---

---

---

---

---


---

---

---


# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

  
software architecture and development

### Support for Batching

- Batching is the process of browsing a large result set in small, reasonable chunks
- The primary goal of Batching is to save system resources (db server load, network bandwidth, client memory)
- Batching has been a major design goal for SmartDataObjects (RowsToBatch property)
- Batching is a major feature of the ProBindingSource object as well

DEV-40: Using SmartDataObjects with the Advanced GUI22

---

---


---

---

---

---


---

  
software architecture and development

### Enabling Batching in the SDO

- Enabling Batching in SmartDataObjects is easy
- Just set the **RowsToBatch** property to a value greater than 0 (200 is the default)
- While opening the query (i.e. during initialization) the SDO will read the first chunk of rows and enable fetching of the next chunks on request

```
{set OpenOnInit True hSDO} .  
{set RowsToBatch 75 hSDO} .  
  
RUN initializeObject IN hSDO .
```

DEV-40: Using SmartDataObjects with the Advanced GUI23

---

---


---

---

---

---

---


  
software architecture and development

### Enabling Batching in the Binding Source

- Enabling Batching in ProBindingSource is easy
- Just set the **Batching** Property to True

```
bindingSource1:HANDLE = hDataQuery .  
  
bindingSource1:Batching = TRUE .
```

- The ProBindingSource will now invoke the **OffEnd** event handler when the UltraGrid control requests missing data (at the 76. row)

DEV-40: Using SmartDataObjects with the Advanced GUI24

---

---

---

---


---

---

---

# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner




software architecture and development

### Reading the next chunks on OffEnd

- The OffEnd event is invoked when the UltraGrid requests more data from the ProBindingSource
- The SDO can read additional data while the property LastRowNum is ? (not yet at the last chunk of data)
- The fetchBatch method of the SDO reads the next batch (SDO maintains all context!)
- ProBindingSource needs to know the number of records added
- Set RowsAdded property of the event argument

DEV-40: Using SmartDataObjects with the Advanced GUI

25



---

---

---


---

---

---

---

---



software architecture and development

### OffEnd event handler

```
METHOD PRIVATE VOID bindingSource1_OffEnd
( INPUT sender AS System.Object, INPUT args AS Progress.Data.OffEndEventArgs ) :

/* Test for last batch. When LastRowNum is set, the SDO has read
the last batch already. */
(get LastRowNum lastRowNum hSDO) .

IF lastRowNum = ? THEN DO:
  (get LastResultRow cLastResultRow hSDO) .

  ASSIGN iLastRow = INTEGER(ENTRY(1, cLastResultRow, "r")) .

  /* Invoke SDO fetch routine */
  RUN fetchBatch IN hSDO
  ( TRUE ) .

  (get LastResultRow cLastResultRow hSDO) .


  /* Calculate number of rows read */
  ASSIGN iNewLastRow = INTEGER(ENTRY(1, cLastResultRow, "r")) .

  /* Tell binding Source */
  args.RowsAdded = iNewLastRow - iLastRow .

END.
END METHOD.
```

DEV-40: Using SmartDataObjects with the Advanced GUI

26



---

---

---


---

---

---

---

---




software architecture and development

### Demo

- Reading data from the SDO in batches

DEV-40: Using SmartDataObjects with the Advanced GUI

27



---

---

---

---

---


---

---

---


# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

  
software architecture and development

### Enable sorting through Grid events

- Batching requires sorting of data in the SDO
- Most .NET grid controls (like UltraGrid) have built in sort capabilities
- But the grid may have only partial data (Batching)
- Sorting the first 200 rows by CustNum by a different column is nonsense (unpredictable)
- Sort order of character data might be different between database and .NET Controls (Umlaute)
- .NET Controls need to execute SDO logic to enable proper sorting in the database

DEV-40: Using SmartDataObjects with the Advanced GUI28

---

---


---

---

---


---

---

  
software architecture and development

### Enable sorting through Grid events

- Sorting behavior is dependent on the .NET Grid
- Some grids rely on the sort capabilities of the ProBindingSource object (Microsoft's DataGridView)
- This will result in execution of the **SortRequest** event handler of the Binding Source
- The UltraGrid uses it's own event logic to intercept sort requests
- Use the **BeforeSortChange** event handler of the UltraGrid

DEV-40: Using SmartDataObjects with the Advanced GUI29

---

---


---

---


---

---


---

  
software architecture and development

### Demo



- Review sort event handler code
- Demo of sorting in the UltraGrid

DEV-40: Using SmartDataObjects with the Advanced GUI30

---

---

---

---


---

---

---


# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

  
software architecture and development

### Agenda

- Introduction
- SmartDataObjects Reviewed
- Data Binding in the Advanced GUI
- Basic Data Binding with SmartDataObjects
- Controlling Batching and Sorting
- **Updating**
- Summary

DEV-40: Using SmartDataObjects with the Advanced GUI 31 

---

---

---


---

---

---


---

---

  
software architecture and development

### Updating Data in the SDO

- SmartDataObject has strict interfaces for updating data
- Two temp-tables are used internally for before-image and current values: **RowObject** and **RowObjUpd**
- Manipulation of RowObjUpd rows is done internally in the SDO (function **submitRow**)
- Update based just on APIs and modified field lists
- Create and delete have different APIs

DEV-40: Using SmartDataObjects with the Advanced GUI 32 

---

---

---


---

---

---


---

---

  
software architecture and development

### Updating Data in the Binding Source

- Similar to sorting there is no standard behaviour to manage data updates between the Binding Source and connected visual control(s)
- Even with similar concepts, most Grid controls use different events for updates and validation
- The UltraGrid offers two events for updating
  - **BeforeRowUpdate**
  - **AfterRowUpdate**
- **BeforeRowUpdate** is cancelable

DEV-40: Using SmartDataObjects with the Advanced GUI 33 

---

---

---

---

---


---

---

---


# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

  
software architecture and development

### Updating Checklist

- Hook into validation event – before/while control expects data to be changed
- Build list of changed fields by comparing values in the control and the current **RowObject** row
- Call submitRow (RowIdent, ChangedValues)
- In case of an error (validation inside SDO) submitRow returns FALSE
  - Use **showDataMessagesProcedure** to display error message to the user
  - Cancel update event: **e:Cancel = TRUE**

DEV-40: Using SmartDataObjects with the Advanced GUI34

---

---


---

---

---


---

---

  
software architecture and development

### Demo

- Code review, update event handler
- Updating data
- Handling validation error messages

DEV-40: Using SmartDataObjects with the Advanced GUI35

---

---


---

---

---


---

---

  
software architecture and development

### Agenda

- Introduction
- SmartDataObjects Reviewed
- Data Binding in the Advanced GUI
- Basic Data Binding with SmartDataObjects
- Controlling Batching and Sorting
- Updating
- **Summary**

DEV-40: Using SmartDataObjects with the Advanced GUI36

---

---

---

---


---

---

---


# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner

  
software architecture and development

### Summary

- SmartDataObjects are compatible with the Advanced GUI for OpenEdge 10.2A
- Manual coding required
- Most coding reusable in components (OO or procedural)
- SmartDataObjects are still the most up to date data retrieval and business logic component supported by OpenEdge tools!

DEV-40: Using SmartDataObjects with the Advanced GUI 37 

---

---


---

---

---


---

---

  
software architecture and development

### Related sessions

- DEV-6: Introduction to the OpenEdge Advanced GUI, Jim Lundy
- PSDN: Introduction to the OpenEdge Advanced GUI Webinar on Tuesday July 15 on [psdn.com/library/entry.jspa?externalID=5393](http://psdn.com/library/entry.jspa?externalID=5393)

DEV-40: Using SmartDataObjects with the Advanced GUI 38 

---

---


---


---


---

---

---

  
software architecture and development



DEV-40: Using SmartDataObjects with the Advanced GUI 39 

---

---

---

---

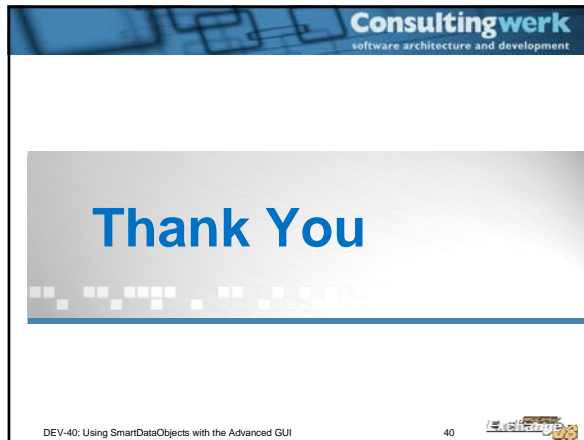
---

---

---

# DEV-40: Using SmartDataObjects with the Advanced GUI

## Mike Fechner



---

---

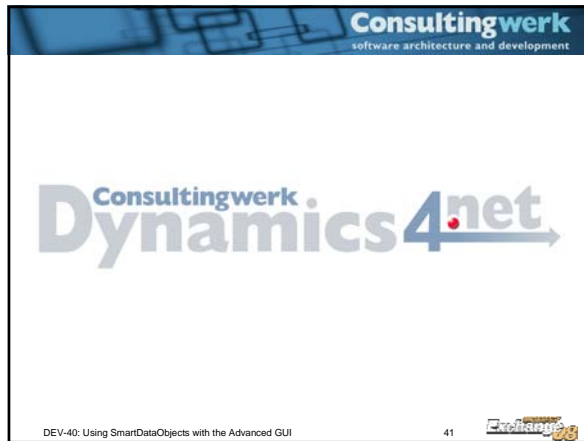
---

---

---

---

---



---

---

---

---

---

---

---