

Implement OERA & achieve true productivity with GUI for .NET™ using SmartComponent Library

Mike Fechner, Director, Consultingwerk Ltd.

Marko Rüterbories, Consultant

info@consultingwerk.de

PUG Challenge Americas 2011, Westford, MA

Consultingwerk
software architecture and development

Be there to win!

All visitors of our booth and attendees of our presentations or workshops that leave a business card or fill out a short form will enter a drawing for an Apple iPod Touch. The lucky winner will be announced at the end of the conference.

- **Getting started with Embedded Windows,** A practical introduction into WinKitLE (practical hands-on workshop), Mike Fechner & Marko Rüterbories, Sunday, June 5th, 1:30 – 4:30
- **SmartComponent Library: GUI for .NET and OERA** (Commercial presentation), Mike Fechner & Marko Rüterbories, Monday, June 6th, 4:00 – 5:00
- **Extending the OpenEdge Architect Visual Designer,** Mike Fechner, Tuesday, June 7th, 4:00 – 5:00
- **Extreme Windows Desktop Integration,** Mike Fechner, Wednesday, June 8th, 11:15 – 12:15

www.consultingwerk.de

Visit us at booth 11

Consultingwerk Ltd.

- Independent IT consulting organization
- Focusing on **OpenEdge** and **.NET**
- Located in Cologne, Germany
- Vendor of tools and consulting programs
- 21 years of Progress experience (V5 ... V10)
- GUI for .NET early adaptor (since 10/2006)
- Just started with iPhone/iPad app development

Consultingwerk Ltd.

- Customers in Germany, Europe, USA
- Working with small to large Progress Partners and direct end users
- Supporting some of the largest Progress Partners in Germany, Belgium, The Netherlands, Austria and UK with application modernization and user interface technologies
- Network of partnering consultants, like ic4b for Web UI's, Whitestar Software, DBAppraise

Solutions for the OpenEdge GUI for .NET

- WinKit
- **SmartComponent Library**
- Dynamics4.NET

- Tools can be used together or separately
- Share common code base

Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- User Interface Design Process
- License model
- SmartComponent Library and WinKit
- Review

Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- User Interface Design Process
- License model
- SmartComponent Library and WinKit
- SmartComponents.Mobile
- Review

Overview

- The out of the box productivity with GUI for .NET is far behind the AppBuilder
- The ***SmartComponent Library*** is our answer to a developers demand for a productive and rich design time experience when using the OpenEdge GUI for .NET

Overview

- A **Component** is a class that provides rich design time experience in the context of a .NET Visual Designer, like the OpenEdge Architect
- Our components are **Smart** by functionality, interoperability and extensibility

At a glance

- Designed to make OpenEdge GUI for .NET developers productive
 - ***UI Design*** and ***Business Logic*** development
- Minimum to no manual code for repetitive tasks
- No constraints in GUI design capabilities
- Rapid prototyping as well as real-world development
- Support past, current and future of OpenEdge
- Part of our ***GUI for .NET adoption strategy***

At a glance

- 99.9% of source code ABL, 0.1% C#
- C# source code required for Visual Designer enhancements

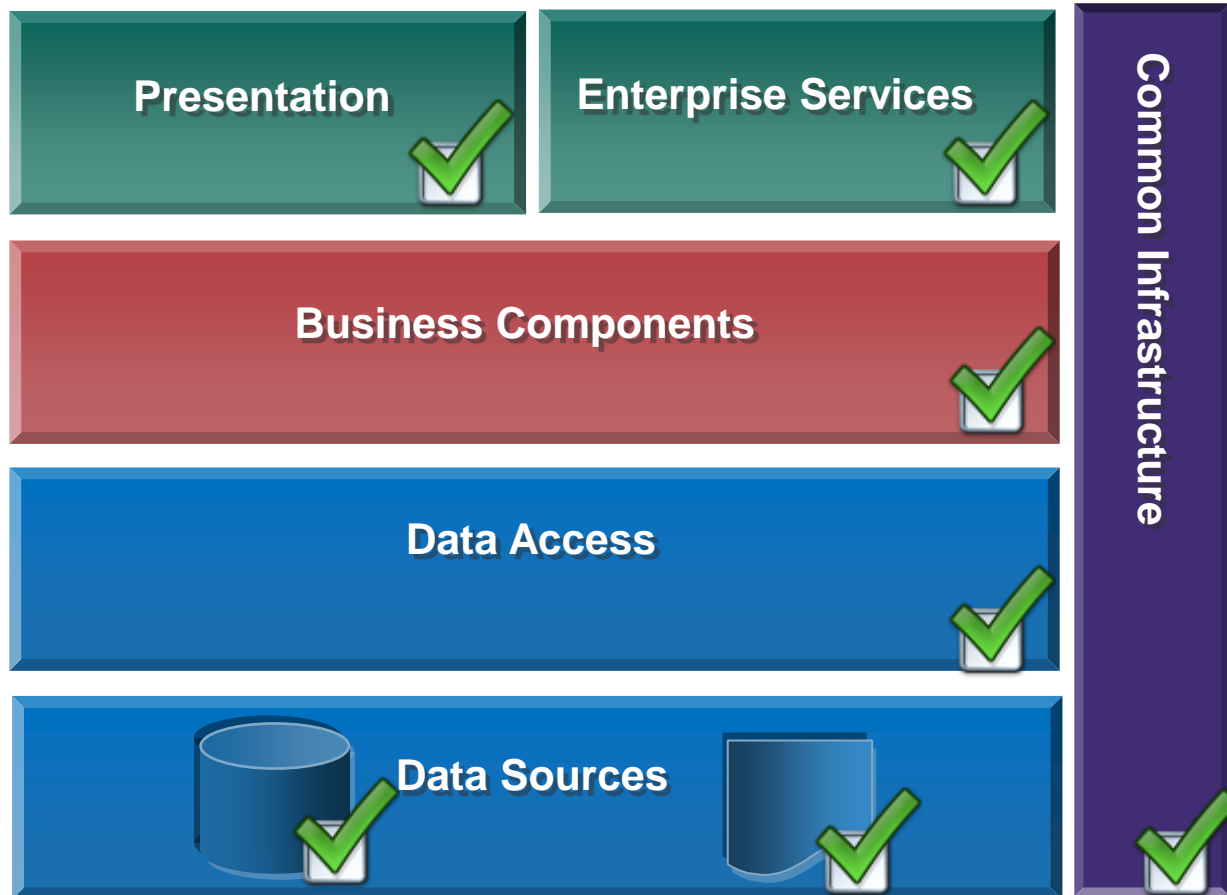
Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- User Interface Design Process
- License model
- SmartComponent Library and WinKit
- SmartComponents.Mobile
- Review

Backend Overview

- The SmartComponent Library provides a flexible OERA Backend Architecture
- Applied OpenEdge Reference Architecture
- Business Entities, Data Access Objects
- Business Tasks, scheduled or async processing
- Common Infrastructure Components
- OERA Backend used for GUI for .NET, ABL and .NET clients, Sonic ESB, Web Services, batch processing, ...
 - write Business Service once, use multiple

OpenEdge Reference Architecture



Alternative frontends for OERA backend

- Alternative frontends (used in production)
 - Web Services
 - ESB
 - Web
- Savvion / OpenEdge BPM
- Mobile devices (native UI, prototype)
 - iPhone, iPad, Android, Windows Phone
 - SmartComponents.Mobile Framework

Frontend Overview

- GUI for .NET client infrastructure
- Extension to OpenEdge UltraControls / Infragistics NetAdvantage for .NET Controls
- Other Control sets may be used
- Integrated into the .NET Visual Designer
 - configuration of behavior and design in a single and intuitive place
 - no restrictions to the way .NET Controls are used
 - wizards and property sheets for repetitive tasks

Frontend Overview

- Specialized application foundation components
 - SmartViewerControl
 - SmartDataBrowser, SmartUpdatableBrowser
 - SmartLookupControl, SmartComboBox
 - SmartToolBarController, SmartPanels
 - SmartWindowForm
 - SmartBindingSource
 - SmartBusinessEntityAdapter
 - Service Adapter, Infrastructure Components

Frontend Overview

- Flexible communication between components
- Defined using Interface types (object orientation)
- Alternative .NET Controls can be integrated with ease

Demo

- Customer Explorer Sample application
- Toolbar / Ribbon
- Customer Detail screen
- Dynamic SmartViewerControl
- Charts, Google Maps
- Various Drag and Drop operations

Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- User Interface Design Process
- License model
- SmartComponent Library and WinKit
- SmartComponents.Mobile
- Review

Connectivity to existing frameworks

- Designed to co-operate with existing frameworks or applications – rather than to compete
- Connectivity by implementation of adapters, Interfaces or hooks
- Proven in customer projects

Connectivity to existing frameworks

- Dedicated interfaces to existing frameworks:
 - Data Access, alternative (OERA) backends, i.e. ticEnterprise
 - Connection management, Service Management
 - Security: Authentication and Authorization, customer frameworks, Progress Dynamics
 - Internationalization
- Overridable functionality:
 - All user interface functionality can be adopted to existing frameworks by Interface implementation or Inheritance / overloading

Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- **Business Logic Design Process**
- User Interface Design Process
- License model
- SmartComponent Library and WinKit
- Review

Business Logic Design Process

- Main Business Logic Components are
 - Business Entities: High level business rules
 - Data Access Objects: Data retrieval, update
- Multiple programming paradigms
 - object-oriented, new development based on 10.2B language features
 - procedural, based on PSDN best practices materials, preferred by some existing developers, still maintained

Business Logic Design Process

- Built around ProDatasets
- Use Data Source objects or your preferred way of populating the ProDataset
- Strict separation of layers:
 - UI design process separated from BL design process
- Business Entities are the foundation of Business Tasks

Business Logic Design Process

- Template based or Visual Design using “Diagrams”
- Flexible source code generator
 - Customizable templates
 - Customizable code generator (ABL class)
- Overridable ABL Business Entity Designer to adopt customer demands, like connection to a data relation repository
- Ad-hoc based Business Entity Test utility

Business Entity Designer

The screenshot displays the Business Entity Designer application window. The main design area shows three entities: eCustomer, eOrder, and eOrderLine. eCustomer has fields: CustNum, Country, Name, Address, Address2, City, State, PostalCode, Contact, Phone, SalesRep, CreditLimit, and Balance. eOrder has fields: Ordernum, CustNum, OrderDate, ShipDate, PromiseDate, Carrier, Instructions, PO, Terms, SalesRep, BillToID, ShipToID, and OrderStatus. eOrderLine has fields: Ordernum, Linenum, Itemnum, Price, Qty, Discount, ExtendedPrice, OrderLineStat, and Indexes (Itemnum, Orderline, OrderLineStat). Relationships include eOrdereCustomerRelation between eCustomer and eOrder, and eOrdereOrderLineRelation between eOrder and eOrderLine.

Data-Relation

Data-Relation	Description
DataRelationName:	eOrdereCustomerRelation
ParentTempTableName:	eOrder
ChildTempTableName:	eCustomer
RelationFields:	CustNum.CustNum
<input type="checkbox"/> Reposition <input type="checkbox"/> Nested <input type="checkbox"/> Foreign Key Hidden	

Field Properties

Field	Description	XML
Name:	Ordernum	
Data Type:	INTEGER	
Initial:	0	
Label:	Order Num	
Format:	zzzzzzzzz9	
Source Field:		

Overview

- eOrder
- eCustomer
- eOrderLine

Demo

- Creation of a Customer-Order-OrderLine-Item Business Entity using the Designer
- Review generated source code
- Review templates / generator / batch generator
- Ad-hoc Query using Business Entity Tester utility

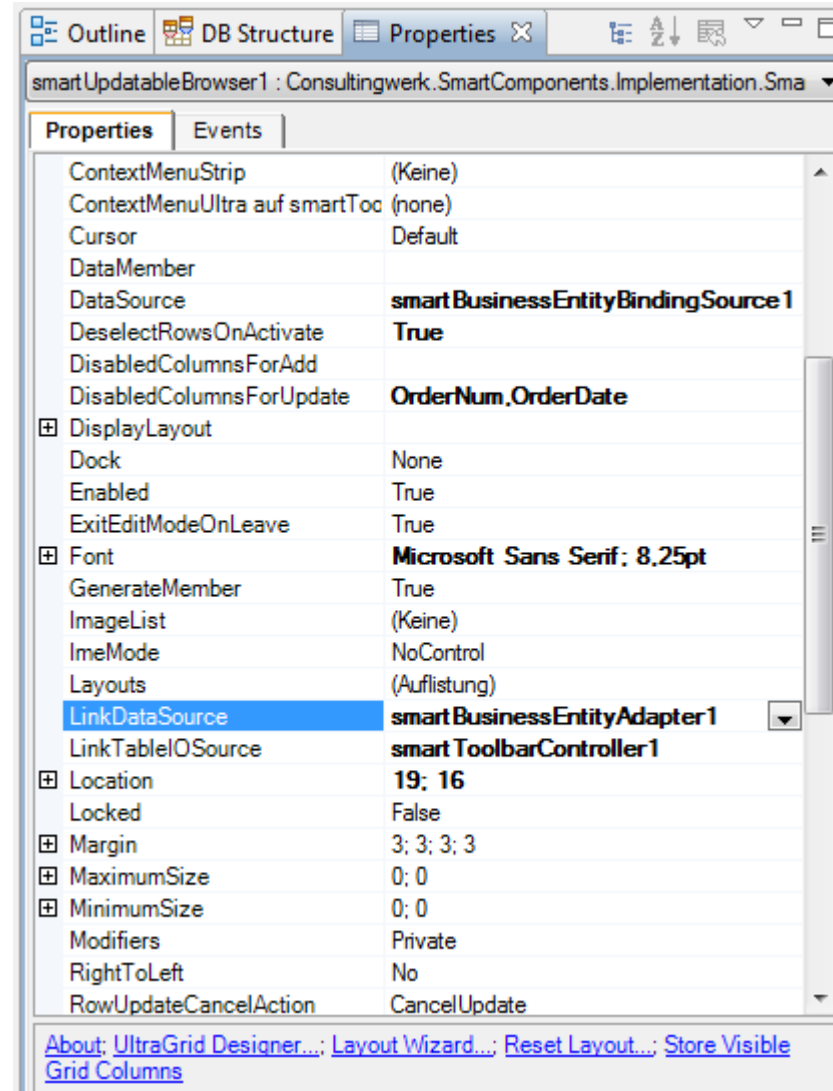
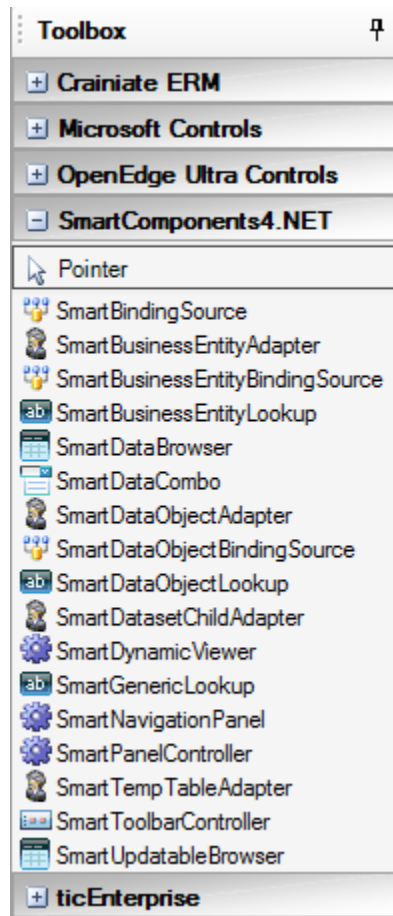
Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- **User Interface Design Process**
- License model
- SmartComponent Library and WinKit
- SmartComponents.Mobile
- Review

User Interface Design Process

- Screen design based on rich foundation classes
- Compose Forms by adding rich SmartComponents and standard .NET Controls
- Connect and parameterize SmartComponents using property grid and Wizards
- Data centric design process: Achieving UI Design productivity similar to the AppBuilder

User Interface Design Process



User Interface Design Process

- Dataset Controller (generated by Business Entity Designer)
- client-side representation of the Business Entity
- optional to use
- ability to reuse client side logic in multiple screens
- key benefit: static access to the ProDataset, no need for dynamic queries on the user interface

Demo

- „Complete“ Order maintenance in 10 minutes
- Order Viewer Design
- Order Line Viewer Design
- Item Number Lookup
- Order Form Design, “putting it all together”

Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- User Interface Design Process
- **License model**
- SmartComponent Library and WinKit
- SmartComponents.Mobile
- Review

License model

- Full source code (99% ABL, 1% C#)
- No runtime royalties
- Corporate license
- First year maintenance is included

- Maintenance renewal from the second year on
- Training and consulting service offerings
- OpenEdge and Infragistics Licenses not included

Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- User Interface Design Process
- License model
- SmartComponent Library and WinKit
- SmartComponents.Mobile
- Review

SmartComponent Library and WinKit

- ***SmartComponent Library*** and ***WinKit*** are part of our GUI for .NET adoption and migration strategy
 - WinKit: Enhancement of existing screens
 - SmartComponent Library: New functionality
- Shared framework foundation classes
- Compatible directory and project structure
- Used side by side in customer projects already

Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- User Interface Design Process
- License model
- SmartComponent Library and WinKit
- **SmartComponents.Mobile**
- Review

Mobile phone apps vs. web apps

- Mobile phone apps offer best „sizzle“ factor for mobile devices
- Gesture recognition (e.g. pull to refresh)
- Control all hardware resources (camera, GPS)
- Store data on devices, cache, offline operation
- Push notification

SmartComponents.Mobile

- SmartComponents.Mobile offers similar API's to developer as on GUI for .NET
- Native UI's on iPhone, iPad (Android, WP7 planned)
- Connected to OpenEdge AppServer using Web Services or FUSE, online or synchronized
- MVC UI Pattern
- Model and Controller platform independent
- View optimized for target device

Demo

- Sports2000 Demo HD
 - iPad App accessing OpenEdge AppServer

Agenda

- Overview
- Architectural Overview
- Connectivity to existing frameworks
- Business Logic Design Process
- User Interface Design Process
- License model
- SmartComponent Library and WinKit
- SmartComponents.Mobile
- Review

Review

- The ***SmartComponent Library*** is our answer to a developers demand for a productive and rich design time experience when using the OpenEdge GUI for .NET
- Based on OpenEdge standard design patterns
- Complete OERA implementation
- Perfectly integrated into the Visual Designer
- **Dramatically reduces training requirements for GUI for .NET**
- Full source code available to developers

Questions



Don't forget to fill out your card!

Consultingwerk
software architecture and development

Be there to win!

All visitors of our booth and attendees of our presentations or workshops that leave a business card or fill out a short form will enter a drawing for an Apple iPod Touch.

The lucky winner will be announced at the end of the conference.

- **Getting started with Embedded Windows,**
A practical introduction into WinKitLE (practical hands-on workshop), Mike Fechner & Marko Rüterborries, Sunday, June 5th, 1:30 – 4:30
- **SmartComponent Library: GUI for .NET and OERA the productive way!**
(Commercial presentation), Mike Fechner & Marko Rüterborries, Monday, June 6th, 4:00 – 5:00
- **Extending the OpenEdge Architect Visual Designer,**
Mike Fechner, Tuesday, June 7th, 4:00 – 5:00
- **Extreme Windows Desktop Integration,**
Mike Fechner, Wednesday, June 8th, 11:15 – 12:15

first name _____
surname _____
e-mail _____
company Name _____

Implement OERA & achieve true product
using **SmartCompo**

Integrate existing app
using **Win**

www.consultingwerk.de

Visit us at booth 11