SmartComponent Library
The OpenEdge Application Modernization Framework

Mike Fechner, Director, Consultingwerk Ltd.
mike.fechner@consultingwerk.de
Consultingwerk Ltd.

- Independent IT consulting organization
- Focusing on OpenEdge and related technology
- Located in Cologne, Germany
- Customers in Europe, North America, Australia and South Africa
- Vendor of tools and consulting programs
- 26 years of Progress experience (V5 ... OE11)
- Specialized in GUI for .NET, OO, Software Architecture, Application Integration
OpenEdge® application modernization solutions

- WinKit
- SmartComponent Library
- Dynamics4.NET

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

- Overview
  - Developing Business Logic
  - Developing GUI for .NET Frontends
  - Telerik Mobile Integration
  - Kendo UI/Angular
  - Migration and Modernization Options
  - Kendo UI Builder
SmartComponent Library

- Developer Framework aimed to increase Developer productivity and flexibility
- Reduce or avoid repeating tasks
  - Tools (code generation and round-trip dev.)
  - Integration with Progress tools (OpenEdge, Telerik, KUIB, BPM, Corticon, …)
  - Application Framework
- Allow integration with existing Applications and Frameworks (3rd party or home grown)
- Allow usage of individual framework components as needed (not all or nothing)
Application Framework Features

- Authentication, including client-principal support and multi-tenancy
- Flexible authorization (menu, toolbar, business logic, custom tokens, ...), can be used for data-authorization (e.g. Regions)
- Localization (full UI translation)
- Definition based referential integrity
- Batch scheduler (repeating, just once, run now)
- Enhanced Unit testing framework
- ...

SmartComponent Library 10
SmartComponent Library

- „eat your own dog food“
- Most of our tools are written using our own libraries and tools, 99.9% ABL code
- Full source code shipped to customers
- Weekly incremental releases (48 releases/year)
- Allow legacy code integration
- Designed for customization
- Don’t understand “best practices” as the “only right way”
OpenEdge Version support

- 10.2B07, 10.2B08
- 11.3.2, 11.3.3 – 11.7
- First day support for 11.6
- Almost first day support for 11.7
- 32 bit/64 bit
- Classic AppServer / WebSpeed
- PASOE
- …
Common Component Specification

- Goal is to define common understanding and architectural patterns to allow interoperability of framework components from different vendors
- Consultingwerk founding member of CCS
- Team leader of Business Entity spec-team
- Containing the term „Component“ in the product name since 9 years – since product launch
- Framework was designed to allow interoperability with 3rd party components
- Framework fully compatible with CCS-ARCH1, CCS-SSS and CCS-BE specs (released in December 2016)
Agenda

- Overview

- Developing Business Logic
  - Developing GUI for .NET Frontends
  - Telerik Mobile Integration
  - Kendo UI/Angular
  - Migration and Modernization Options
  - Kendo UI Builder
Backend Overview

- The SmartComponent Library provides a flexible OERA Backend Architecture
- **OpenEdge Reference Architecture**
- Business Entities, Data Access Objects are a key component
- Business Tasks, including support for scheduled and async processing
- Common Infrastructure Components, Services
Business Logic Design Process

- Template based, Visual Design using Diagrams
- Flexible source code generator
  - Customizable templates
  - Customizable code generator (ABL class)
- Plugin Infrastructure (integrate your own ABL code)
  - Influence automatic naming and design process
  - Add and enhance code generation
- Query custom repositories during design phase
- Ad-hoc based Business Entity Test utility
Demo 1

- Create SalesRep Business Entity
- Add „CustomDescription“ and „AverageQuota“ Calculated Fields
- Review in Progress Developer Studio
Demo 2

- Create Customer Business Entity
- Add Validation on Name
- Query Customer using DatasetModel
- Update using DatasetModel
- Add custom method to Business Entity
  - Put Customer On Hold
- Invoke custom method using Model Method
Business Logic reuse – write once use in:

- OpenEdge GUI for .NET
- OpenEdge GUI/TTY/Batch
- Telerik Kendo UI/Kendo UI Builder
- Telerik Mobile and NativeScript
- WebSpeed
- OpenEdge BPM
- Native Mobile Applications (Xamarin Studio)
- Progress Rollbase
- Web Services and REST Services
- Sonic ESB
- DataDirect OpenAccess SDK
- Unit Testing frameworks
Agenda

- Overview
- Developing Business Logic
- Developing GUI for .NET Frontends
  - Telerik Mobile Integration
  - Kendo UI/Angular
  - Migration and Modernization Options
  - Kendo UI Builder
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

[Diagram of user interface design process with SmartComponent Library]
Demo

- Use Business Entity Designer to generate UI
- Open in Visual Designer
- Add Salesrep Lookup in Visual Designer
Agenda

- Overview
- Developing Business Logic
- Developing GUI for .NET Frontends
- Telerik Mobile Integration
  - Kendo UI/Angular
  - Migration and Modernization Options
  - Kendo UI Builder
Support for Progress/Telerik Mobile

- SmartComponent Library Backend and Progress/Telerik Mobile and Web Options = perfect match!

- Full support for complete JSDO functionality

JSDO Consultingwerk likes this.
Support for Progress/Telerik Mobile

- Generic REST Service Interface for JSDO based clients
  - Full CRUD support
  - Invoke Methods
  -Invoke Business Tasks
- No need to redeploy new REST application for every single application
- Direct access to any Business Entity or Business Task that the current user is authorized for
Generic REST Service Interface for JSDO

- Simplified deployment
- PASOE (Progress AppServer) or Classic AppServer
- Web Handlers (OpenEdge 11.6), PASOE preferred
- REST Adapter (OpenEdge 11.5, Classic AppServer supported
- Single developer AND large teams
- Translation of Data Catalog
Demo Telerik Mobile Platform

- Create a new hybrid mobile project in Telerik Mobile platform using Wizard
- Add access to SalesRep Business Entity
- Leverage same Business Entity in Web and Desktop

- Recorded version of the demo: https://www.youtube.com/watch?v=m-63Wzpjs9c
Agenda

- Overview
- Developing Business Logic
- Developing GUI for .NET Frontends
- Telerik Mobile Integration
- Kendo UI/Angular
  - Migration and Modernization Options
  - Kendo UI Builder
SmartComponents.Web2

- Angular is a MVVM framework for browser based applications
- Binding Data to UI Elements
- Separate Data, View and Code
- Kendo UI provides UI components for Angular
- SmartComponents.Web2 implements similar programming model as in GUI for .NET
- Configurable data sources and communication between UI components
Demo

```html
<div ng-controller="simpleCtrl" style="background-color: white;">
  <h1>Salesreps</h1>
  <div class="row">
    <div class="col-md-28">
      <smart-data-source
        smart-object-name='salesrepDataSource'
        smart-entity-table="eSalesrep"></smart-data-source>
      <smart-grid
        smart-object-name='salesrepGrid'
        smart-data-source='salesrepDataSource'
    </div>
  </div>
  <div class="col-md-20">
    <smart-toolbar
      smart-object-name='salesrepToolbar'></smart-toolbar>
    <smart-viewer
      smart-object-name='salesrepViewer'
      smart-data-source='salesrepDataSource'
      smart-tableio-source='salesrepToolbar'
  </div>
</div>
```
Demo Kendo UI

- AppServer Backend
- Generate Kendo UI Grid Screen
- Server-side filtering, sorting
Progress UI flexibility strategy

- SmartComponents.Web2 built to fit with Progress UI strategy
- Built around UI meta data
  - Annotations in code
  - Repository Database
  - Generated code
- Common Component Specification compliant
- Progress Kendo UI Builder integrated
SmartComponents.Web2

- Initially Based on Angular JS 1 (JavaScript)
- Rewritten in Angular 2 (November 2016 – June 2017)
- Complete rewrite in TypeScript as a OO solution
- Leveraging communication protocols known from SmartComponent Library for GUI for .NET
Agenda

- Overview
- Developing Business Logic
- Developing GUI for .NET Frontends
- Telerik Mobile Integration
- Kendo UI/Angular
- Migration and Modernization Options
- Kendo UI Builder
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
- Over 40 different WinKit implementations already
- WinKit included in SmartComponent Library
### Order Maintenance

**Order Num:** 1  
**Ordered:** 23/01/09  
**Order Status:** Shipped

- **Shipped:** 28/01/2009  
- **Promised:** 28/01/2009  
- **Cust Num:** 53  
- **Name:** Offside Hockey  
- **Carrier:** FlyByNight Courier  
- **Terms:** Net30  
- **PO:**

#### OrderLines

<table>
<thead>
<tr>
<th>Line Num</th>
<th>Item Num</th>
<th>Item Name</th>
<th>Qty</th>
<th>Price</th>
<th>Extended Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>54</td>
<td>Canoe</td>
<td>30</td>
<td>4.86</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>55</td>
<td>Surfboard</td>
<td>21</td>
<td>23.85</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>53</td>
<td>Swimming Trunks</td>
<td>75</td>
<td>8.77</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>28</td>
<td>Ski Gloves</td>
<td>39</td>
<td>23.00</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>39</td>
<td>Baseball</td>
<td>75</td>
<td>34.00</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>26</td>
<td>Catchers Mit</td>
<td>28</td>
<td>4.99</td>
<td></td>
</tr>
</tbody>
</table>

**Line Num:** 1  
**Item Num:** 54  
**Item Name:** Canoe  
**Price:** 4.86  
**Qty:** 30  
**Extended Price:** 131.22  
**Discount:** 10%  
**Order Line Status:** Shipped
Migration and Modernization options

- **Enhance and extend existing applications**
- SmartComponent Library integrates with existing applications and frameworks
- SmartComponent Library functionality can be accessed from existing code
- SmartComponent Library can access existing code, typically of any kind
  - direct execution or through adapters
  - custom service or component implementations
  - Don’t worry about SHARED VARIABLES etc…
Alternative

- Migration of existing Applications
Migration and Modernization options

- Migration of partial source code
- Options vary based on current code
- Strong usage of Proparse for analyzing source code, importing into Business Entity Designer
- Migration of any kind of applications
  - custom
  - ADM1, ADM2 or Dynamics
- Template based migration of UI components
- API for ABL Widget abstraction
Demo

- Migration of ABL Browse Widget
  - Rewrite of Calculated Field
  - Creation of matching Business Entity
- Migration of ABL GUI for .NET Frame
  - Implement ABL Widget abstraction
- Migration to Angular View
Agenda

- Overview
- Developing Business Logic
- Developing GUI for .NET Frontends
- Telerik Mobile Integration
- Kendo UI/Angular
- Migration and Modernization Options
  - Kendo UI Builder
Kendo UI Builder

- Graphical Designer for Kendo UI Frontends
- Leverages JSDO and Data Object Catalog
- Application Generation/Build Framework
- Angular JS and responsive UI
Support for Kendo UI Builder

- Business Entity Designer provides support for Semantic Type
- Generation of meta data for Data Provider
- Generation of meta data for User Interface
- Migration of ABL GUI to KUIB meta data for Views
- Extensive ABL API for manipulating KUIB meta data
- Integration of components (like Lookup) from SmartComponents.Web2
Questions