SmartComponent Library
by Consultingwerk
software architecture and development

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

- Independent IT consulting organization
- Focusing on **OpenEdge** and **related technology**
- Located in Cologn, Germany, subsidiary in UK
- Customers in Europe, North America, Australia and South Africa
- Vendor of developer tools and consulting services
- 27 years of Progress experience (V5 … OE11)
- Specialized in GUI for .NET, Angular, OO, Software Architecture, Application Integration

http://www.consultingwerk.de/
Agenda

- Overview
  - Framework Backend Architecture
  - OpenEdge GUI for .NET Frontends
  - Kendo UI and Angular Frontends
  - NativeScript Clients
  - Application Modernization
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)
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 (45 releases/year)
- Technical support
- Support for new OpenEdge releases in no longer than 30 days
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
- …
Agenda

- Overview
- **Framework Backend Architecture**
  - OpenEdge GUI for .NET Frontends
  - Kendo UI and Angular Frontends
  - NativeScript Clients
  - Application Modernization
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
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)
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

- Build Customer Business Entity
  - Validation
  - Calculated Field
- Business Entity Tester
- Use DatasetModel to access Business Entity
RESTful

- Standard RESTful interfaces build into framework
- No custom code required
- Web Developer friendly
- Foundation for API based economy
- Support smart devices like Google Home and Amazon Echo
- Swagger based API documentation
Demo

- Enable RESTful access to Customer Business Entity
- [http://localhost:8820/web/Entities/DemoCustomers](http://localhost:8820/web/Entities/DemoCustomers)
SmartFramework ERD Designer

- Available since 11/2017
- Based on Zalmoxis ERD
- Integration with SmartFramework backend through AppServer
Demo

- Import Sports2000 Schema into ERD
- Import Relation Definition from SmartFramework backend
Agenda

- Overview
- Framework Backend Architecture
  - OpenEdge GUI for .NET Frontends
  - Kendo UI and Angular Frontends
  - NativeScript Clients
  - Application Modernization
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
Demo

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

- Overview
- Framework Backend Architecture
- OpenEdge GUI for .NET Frontends
  - Kendo UI and Angular Frontends
- NativeScript Clients
- Application Modernization
Support for JSDO Based Clients

- SmartComponent Library Backend and
  JSDO based consumers: Telerik Mobile, Kendo UI, Kendo UI Builder, … = 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
Recorded 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
SmartComponents.Web2

Kendo UI
THE ART OF WEB DEVELOPMENT

Angular
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
Screen Definition based on functional components

```html
<div ng-controller="simpleCtrl" style="background-color: white;">
  <h1>Salesrep</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-toolbar-source='salesrepToolbar'
  </div>
</div>
```
Demo

- SmartComponents.NG2 sample application
- Screen definition
Progress UI flexibility strategy

- SmartComponents.NG2 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 – September 2017)
- Complete rewrite in TypeScript as a OO solution
- Leveraging communication protocols known from SmartComponent Library for GUI for .NET
Agenda

- Overview
- Framework Backend Architecture
- OpenEdge GUI for .NET Frontends
- Kendo UI and Angular Frontends
- **NativeScript Clients**
- Application Modernization
NativeScript

- Native mobile applications for iOS and Android
- JavaScript or TypeScript based
- Standard for Angular based mobile applications
- Allows us to reuse parts of SmartComponents.NG2 on mobile
- Similar developer experience for customers
- Similar component architecture
  - Data sources, grids, viewer, lookups, …
Agenda

- Overview
- Framework Backend Architecture
- OpenEdge GUI for .NET Frontends
- Kendo UI and Angular Frontends
- NativeScript Clients

- Application Modernization
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 45 different WinKit implementations already
- WinKit included in SmartComponent Library
Order Maintenance

Order Num: 1  
Ordered: 23/01/09

Order Status: Shipped

Shipped: 24/01/09

Purchased: 25/01/09

Cust Num: 58  
Name: Offside Hockey

Carrier: FlyByNight Couriers  
Terms: Net30  
P.O.: 

Order Lines

<table>
<thead>
<tr>
<th>Line Num</th>
<th>Item Num</th>
<th>Item Name</th>
<th>Qty</th>
<th>Price Extede</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>56</td>
<td>Canoe</td>
<td>30</td>
<td>4.86</td>
</tr>
<tr>
<td>2</td>
<td>55</td>
<td>Surfboard</td>
<td>21</td>
<td>20.96</td>
</tr>
<tr>
<td>3</td>
<td>53</td>
<td>Swimming Trunks</td>
<td>75</td>
<td>8.77</td>
</tr>
<tr>
<td>4</td>
<td>28</td>
<td>Ski Gloves</td>
<td>39</td>
<td>23.00</td>
</tr>
<tr>
<td>5</td>
<td>39</td>
<td>Baseball</td>
<td>75</td>
<td>34.00</td>
</tr>
<tr>
<td>6</td>
<td>26</td>
<td>Catchers Kit</td>
<td>28</td>
<td>4.38</td>
</tr>
</tbody>
</table>

Line Num: 1

Item Num: 56

Item Name: Canoe

Price: $4.86

Qty: 30

Extended Price: $111.22

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

- Semi-automated Migration of source code
- Options vary based on current code
- Strong usage of Proparse for analyzing source code, importing into Business Entity Designer
- Refactoring tooling based on Proparse
- 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 TTY style application with UPDATE EDITING Block
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
Questions