

Dynamics4.NET



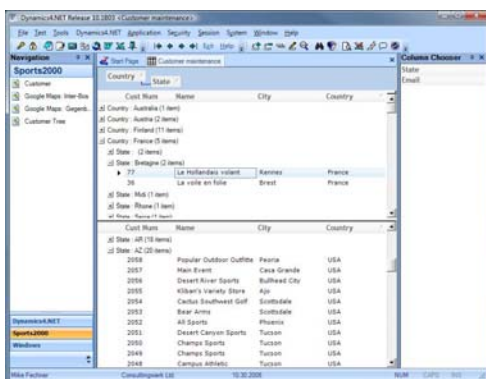
Dynamics4.NET is the fastest way to migrate your Dynamics or ADM2 application to the “OpenEdge GUI for .NET”. Gain advantage of the competitive user interface of the next OpenEdge release 10.2A without affecting your business logic and with minimum changes to your development process.

Consultingwerk Ltd. located in Cologne, Germany has developed a new rendering engine for Progress Dynamics (aka OpenEdge Studio Framework) based applications. This rendering engine is based on the “OpenEdge GUI for .NET” and offers a high level of compatibility with existing Dynamics applications. With our development we are focusing on the migration of existing Dynamics applications, their maintenance and new developments.

Key benefits

- Protect investment made in Dynamics development
- Improve competitiveness with a state of the art user interface
- ADM2 based rendering engine, offering maximum compatibility with existing Dynamics applications
- Migration to the new OpenEdge GUI for .NET by changing attributes in the repository using migration utilities instead of manually changing UI design or rewriting code
- “Mix and match” operation with the classic Dynamics rendering engine, including links between .NET and Progress UI based windows
- Container support: Dynamic object controller, folder window, treeview container and dynamic SmartFrame
- MDI container support (optional), MDI Menu controller
- SmartObject support: DynBrowser, DynViewer, DynToolbar, ...
- SmartDataField support: DynLookup, DynCombo, ...
- Dynamic rendering or static code generation (PGEN)
- Implementation of the client logic API provides abstract access to .NET controls from client side super procedures
- Compatibility layer eases migration of custom code that does not use the client logic API

Feature details

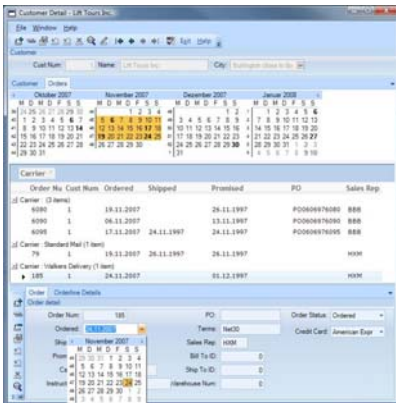


The rendering engine enhances the appearance of a Dynamics application with a modern .NET based user interface. Dockable panes, MDI support, graphical toolbar objects and explorer bars are provided to improve the presentation layer. Infragistics controls enhance the user experience when working in input forms (e.g. calendar or calculator controls). Multiline layout, card views, etc. enhance the visual appearance of browsers in the application. Modern control sets like calendars or graphs can be utilized to display and edit data in a more natural and advanced way. Drag and drop can be used for filtering data or as an event to your business logic to update information as required.

The .NET resizing supports the usage on screens of virtually any resolution. The resizing is controlled using native .NET WinForms resizing attributes stored in the Dynamics repository (Dock, Anchor) and are supported by most .NET controls. .NET controls enable a much faster rendering process and smoother and faster resizing than the previously used Progress widgets.

Customizing, translation and security are supported by the Dynamics4.NET rendering engine as they are supported by the standard rendering engine.

Architectural overview



The rendering engine is based on the most recent release of Progress Dynamics. Compatibility is achieved by making only minimal changes to the repository manager and required Dynamics components. The visible components of the ADM2 and rendering objects are rewritten using a compatibility layer. An alternative set of rendering objects uses standard repository definitions based on .NET controls. The compatibility layer is built on ABL classes and custom .NET controls that close the functional gap between .NET controls and ABL widgets. Standard Dynamics components like the filter screen, the message dialog or the login prompt are rewritten using the same method. The client logic API and the compatibility layer support developers of client logic by only requiring the minimum set of changes to existing code. Business logic is not affected by the changes at all.

Dynamics manager access has been OO enabled. Thus the usage of the manager's functionality is simplified even when designing additional static components using the Visual Designer of OpenEdge 10.2A.

Migration

Migration of existing applications requires changing the rendering engine for each affected object by changing the object type (i.e. from DynObjc to DynObjc.NET, DynView to DynView.NET). This change can be done manually using standard Dynamics tools or automated for a complete container or a set of containers using our migration utilities. These tools also support the creation of copies of nested objects used in instances. This is very useful when a viewer is used in a container that is already migrated and in a container that may not (yet) be migrated.

Dynamics4.NET is flexible enough to leverage customizations to an existing Dynamics and ADM2 environment.

An initial migration should be done together with our consultants in a 5 day workshop. This workshop covers training, the initial migration of (parts of) the .NET features will enhance the user experience even more.

Contact information

For further information or presentation requests please contact:

Consultingwerk Ltd.
Attn.: Mike Fechner
Unter Käster 1-3
50667 Köln
Germany

www.consultingwerk.de
info@consultingwerk.de

Fon.: +49 (0) 221 / 27 60 9-40
Fax.: +49 (0) 221 / 27 60 9-41