Web Dynpro Know-How That Every Developer Needs to Enhance SAP Solutions

Karl Kessler, SAP AG

Much attention has been given to SAP’s ERP strategy and all the benefits it delivers to customers — including bringing continuous innovation to ERP processes without requiring frequent, full-blown application upgrades.

As part of this strategy, SAP ERP 6.0 is built on a stable application core based on SAP NetWeaver 7.0, and innovation will be delivered in regular enhancement packages (EhPs) for SAP ERP. The enhancement package procedure significantly reduces the efforts previously spent on SAP ERP upgrade projects to introduce business process innovation.

The first enhancement packages are already available: Enhancement package 2 is out now, and enhancement package 3 has gone to Ramp-Up as of this writing. New enhancement packages are planned roughly every nine months.

For developers, there is some additional information — beyond these EhP basics — that you should know:

- Enhancement packages contain new and simplified business processes with user interfaces (UIs) implemented in Web Dynpro and running in SAP’s enterprise portal.
- Enhancement packages deliver content for the Enterprise Services Repository in the form of enterprise services bundles, published at http://sdn.sap.com. In many cases, process innovations are achieved by delivering a UI based on Web Dynpro and an underlying service-enabled business logic.
- Technically, enhancement packages are based on the switch and enhancement framework — a generic mechanism that allows you to selectively activate certain elements of applications, such as the UI, as needed. The framework resides in the ABAP stack of SAP NetWeaver.

- New industry solutions for SAP ERP use enhancement packages to deliver industry-specific business process extensions required for a particular industry.
- Enhancement packages are easy to install and activate. They are purely optional and are implemented by the customer on an as-needed basis.
- All other offerings in SAP Business Suite plan to follow SAP ERP in its enhancement package approach.

SAP’s ERP strategy offers SAP customers the choice of when to implement the optimal roadmap for innovation in their system landscape and processes. Smooth introduction of any innovation is of paramount importance in mission-critical environments such as SAP ERP; enhancement packages allow customers to schedule innovation at their own pace, not forced by technical implementation constraints.

Developers, then, need to understand Web Dynpro, the switch and enhancement framework, and how to apply enhancements — including those introduced by enhancement packages — to their current solutions. In this column, I’ll give a high-level introduction to enhancing a simple Web Dynpro application through one of these implicit extension points. This will show both the power of Web Dynpro and the corresponding development tools of the ABAP Workbench, and you will see how easy it is to develop in the ABAP Workbench’s enhancement mode.


The Technology and Tools You’ll Need to Enhance Web Dynpro Applications

A product strategy like the one described earlier has two requirements for developers:

- A solid technical basis that gives developers the flexibility to make adaptations needed by the upper layers of an application, such as SAP’s ERP business processes
- Strong tool support for application developers and the administrators responsible for supporting those applications

Web Dynpro is this technical basis. And the tools in the ABAP Workbench offer straightforward support for this approach while enhancing the newly delivered Web Dynpro applications that are at the core of SAP’s enhancement package strategy for SAP ERP and SAP Business Suite.

In addition, the switch and enhancement framework is the key to making applications flexible and extensible. It allows you to add innovation to applications with minimal disruption of the original application code. You can also set up a “switch” to your enhancement, allowing you to activate it as needed and, in some cases, revert to the original if the enhancement is no longer required (provided there are no dictionary changes).

While this flexibility is most evident when adding UI features like a field or a customer subscreen, it extends to all layers of a typical application: the Web UI, the business logic, and the database layer. On each level, the tools of the ABAP Workbench provide built-in support to help SAP developers design flexible applications — and to later help developers extend and adapt those applications.

Some Considerations: Enhancements vs. Code Modifications

This flexibility becomes increasingly important when, for example, SAP develops a generic core application, then adapts it for a particular industry flavor, then later offers a specific country version — all before the SAP customer adds any enhancements of its own.

If you have modified the code — intertwined your own custom code with SAP’s original code — you will inevitably face long cycles of conflict resolution when you need to upgrade, when there are enhancement packages or future enhancements, or when you do any troubleshooting.

Instead, the new enhancement package approach relies on enhancements, where the developer starts to extend an existing application rather than make direct modifications to the original application code.

Look for Explicit Extension Points

Often enhancements are done through explicit extension points carefully designed by the original application provider. In these cases, it’s clear where you can make your extensions. The Workbench includes a switch and enhancement framework infosystem that shows you all potential extension points inside the code where you can start your enhancement work.

Or, Use Web Dynpro to Create Your Own Extension Points

In other cases, you may require implicit extension points that are supported through the underlying Web Dynpro programming model. (This article will cover this process of adding your own enhancements.)

In Either Case, Separate Your Enhancements from the Original Code

Whichever your approach, the most important recommendation when enhancing an application — be it Web Dynpro or any classical user interface — is to keep any extensions separate from the original code as much as possible, and add extensions only at a few well-chosen, clearly indicated points.

The reason is obvious: When it comes to maintenance, enhancing an application is a far better approach than traditional modifications. What’s more, the original code might change over time in subsequent releases or support packages. By keeping your code in separate “containers” on your side and maintaining a few easy-to-identify enhancement points, future maintenance is easier. If you follow this discipline, you will much better manage your enhancement code. (Needless to say, SAP follows a similar approach when it enhances the generic core application to implement a particular industry flavor.)

Let’s go through a simple example that covers the basics of enhancing a Web Dynpro application through an implicit extension point. While this article won’t go into detailed instructions on creating a Web Dynpro enhancement, it will give you an overview of what you’ll see as you make an enhancement to a Web Dynpro application. For more general information on creating a Web Dynpro application, please see the Additional Resources at the end of this article.
An Overview: How to Enhance a Web Dynpro Application

Figure 1 shows a sample Web Dynpro application: the ubiquitous flight search demo application along with its enhancement, a Flight Bookings table. (While this is not an enhancement package implementation, you can apply the basic principle here to future custom enhancements as well as enhancement package implementations.) The original FLIGHTS_DEMO application in Figure 1 consists of four search input fields where you can specify the search criteria for airline flights: the departure city and country and the destination city and country. Hit the Select Flights button, and the system will search for corresponding flights and display the results in the Flight List table.

In the original application, all search fields come with value help, and selecting F4 brings up available flight destinations. Web Dynpro is also clever enough to automatically fill the country field as you enter your destination city, since the field is derived from a dictionary table.

The development of such a Web Dynpro application is straightforward using the ABAP Workbench, the Web Dynpro Builder, and the accompanying wizards, tools, and features that simplify the process of designing these Web-based applications with drag-and-drop ease.

Designing the Enhancement

Now, let's get started with the enhancement itself. We want to enhance the flight application by adding a way for a user to see all current bookings for a selected flight. We will enhance the original view with a button named Get Bookings that retrieves booking information (table SBOOK) for the selected flight (shown in Figure 1) and displays it in a custom view.

In the design layer, this requires only a small change to the original application — adding a button that navigates to a custom view. Even if the layout or flow of the original application changes, it is then easy to reinsert the button at the proper place. The more complex features of this custom view (e.g., retrieving the booking information or populating a corresponding Web Dynpro table with found bookings) are all kept apart from the original application.

Figure 2 on the next page shows the original application in the Workbench. By selecting the spiral icon button (upper left), we enter the enhancement mode. On the left is the outline of the Web Dynpro component. On the right is the WYSIWYG layout of the search and display view.

How the Enhancement Mode Simplifies Your Work

In enhancement mode, the system first prompts you to name an Enhancement Implementation, which we call FLIGHT_DEMO_ENH. This will serve as the container for all your enhancement artifacts, keeping them separate from the core application code. Later, you

3 Of course, if the application is not changed with the next release cycle, no action is necessary.
have the option to aggregate multiple enhancement implementations to a single composite enhancement implementation.

In enhancement mode, you can view most elements of the original application as display-only, while still being able to enhance (or add) custom elements. (You can also hide certain pieces of the original application from view.)

In Figure 3, we add our custom Get Bookings button by simply appending a button control from the context menu to the original control tree displayed in the upper right. The control tree shows all controls and their corresponding control properties in the lower right corner. If a custom control has been added, the enhancement control container is referenced in the control’s properties (lower right). Then we add an outbound plug to connect the original view to a not-yet-existing custom view. Generally, the WYSIWYG layout for the enhancement behaves just as it would if you were normally creating or changing a Web Dynpro application.

The ease of the Web Dynpro programming model allows developers to connect one application view (flight display) to the other (bookings) using the Get Bookings button that we just defined. The Web Dynpro code wizard automatically adds the required code for navigation between the two views. This becomes the enhancement point to the original application.

At this point, you’d create the custom view for the bookings screen as you would with any Web Dynpro...
application. To view the application at the point that you have defined the button, created the bookings view, added two plugs (one inbound for the bookings and one outbound to the display view), and connected them, see the view hierarchy displayed in the ABAP Workbench, as shown in Figure 4. In this screen, you see the original Web Dynpro application in the upper left of the navigation pane. The booking screen is kept separate from the Enhancement Implementations folder. The views and their plugs and connections are shown in a tree hierarchy on the right — much like a standard Web Dynpro application. If you select a custom plug from this structure, its properties and the associated enhancement container are shown at the bottom of the screen (see Figure 4).

The rest of the enhancement project is pretty straightforward, typical of any Web Dynpro application development: You must implement the custom view by creating a Web Dynpro table that contains the bookings, and then maintain the underlying Web Dynpro context that stores current bookings based on the selected flight.

Later, once you have undertaken several enhancement implementation projects, you may find it helpful to have a view of the enhancements in your system. You’ll get this overview by selecting Enhancement Info System (highlighted in the top left of Figure 4). This shows both the enhancement definitions and implementations available, and allows you to navigate to the affected development objects so that you can inspect them.

**Implications for Development and Innovation in Your Current Solutions**

The same approach and tools that allow you to build powerful Web-based applications are also the ones you can use to enhance your applications — and at the same time, help ensure a sustainable, long-term strategy for solution management.

By relying on enhancement points rather than on code modifications, maintenance efforts can become more efficient and business processes more easily adaptable. Web Dynpro’s enhancement mode allows you to keep your enhancements separate from the original application by adding implicit enhancement points that are easy to reinsert once the original is changed. You can also switch these enhancements on or off as they are needed.

You can easily access and review your enhancements now and into the future, when larger-scale maintenance or improvements are required, to help ensure the continuity of your business processes while leaving your solutions open to innovation.

For more information on Web Dynpro development, visit [http://sdn.sap.com → SAP NetWeaver Capabilities → Development and Composition → ABAP](http://sdn.sap.com). Here you will find abundant Web Dynpro ABAP and switch and enhancement framework resources. And for more on enhancement packages, please visit [http://service.sap.com/erp-ehp](http://service.sap.com/erp-ehp).

---

**Additional Resources...**

- “Get started developing Web-native custom SAP applications with Web Dynpro for ABAP” by Ulli Hoffmann ([SAP Professional Journal](http://www.SAPpro.com), July/August 2007, [www.SAPpro.com](http://www.SAPpro.com))
- “Introducing the switch and enhancement framework – consolidating industry solutions with the mySAP ERP core” by Karl Kessler ([SAP Professional Journal](http://www.SAPpro.com), March/April 2006, [www.SAPpro.com](http://www.SAPpro.com))
- Web Dynpro for ABAP by Ulli Hoffmann (SAP PRESS, [www.sap-press.com](http://www.sap-press.com))