Abstract— This work is presenting a study performed towards the web programming on integrated system, emphasizing the facilities offered by the SAP integrated system’s environment. Because the architecture of the SAP NetWeaver Application Server system is a three-tier architecture, it’s possible to develop web business applications of ABAP (Advanced Business Application Programming) and/or Java type. The web applications for the preformed study were implemented on the trial version SAP NetWeaver Application Server and programmed by means of ABAP language. For achievement, it’s been chosen one of the most advanced technologies, the faceless component-making technology.

I. INTRODUCTION

A. The three-tier SAP NetWeaver Application Server Architecture

The SAP NetWeaver Application Server represents the central part of the SAP software stack that makes possible the development of ABAP and/or Java applications for data storing and processing. It completely supports the J2EE (Java 2 Enterprise Edition Platform) standard. The SAP NetWeaver Application Server is developed for Web applications, as well.

As explained in [1], the SAP NetWeaver AS architecture is structured on three levels (three-tier architecture) Fig. 1: persistence level-1, application level-2, and presentation level-3.

When a web browser is used in the application programs, then can be accessed web pages realised in the presentation level. The application programs access these web pages through the ICF (Internet Connection Framework) classes and interfaces. The presentation in web browser is based on the Web Dynpro ABAP technology and Business Server Page (BSP), similar with Java Server Script (JSP), the predecessor of the Web Dynpro ABAP technology. The presentation in SAP GUI is based on the classical Dynpro technology.

The application level, is the software level where the ABAP and/or Java programs are executed. It is the level made of a kernel written in C/C++ that serves as a platform for the application programs, independent of the hardware, operation system and databases, and provides the processors (virtual machines) for the future programming languages: ABAP with ABAP Objects, XSLT (Extensible Stylesheet Language Transformation) and ST (Simple Transformation), for transforming the data in XML format.

The persistence level, is the level where the persistent data are kept in databases. The central database can be accessed through the interface of the AS ABAP database, made of two interfaces: Open SQL and Native SQL. The databases that can be integrated are: Oracle, MS SQL Server, IBM DB2 Universal Database for Unix and Windows, SAP liveCache technology MaxDB, IBM DB2 Universal Database for z/OS, IBM DB2 Universal Database for iSeries and IBM Informix. These three levels are described explicitly by Ulli Hoffmann in his book [2].

Communication between the three levels is presented in [1], through their corresponding interfaces: the Internet Communication Manager (ICM) that realises the connexion to Internet, the Remote Call Function (RCF), connexion for the access of external software applications (or another SAP NetWeaver AS system), interfaces for the communication with the persistence level. The ICM enables the direct communication with the Internet through HTTP/HTTP/SMTP, based on presentation components, as Web Dynpro ABAP and BSP.

The AS ABAP can be, in the same time, both client and server for the Web services. The ABAP programs access the ICM through the ICF (Internet Communication Framework) classes and interfaces, as it [3] relates.

The RCF interface can realise the connexion among different AS ABAPs, or between the AS ABAP and an external system (non SAP).
The integrated SAP NetWeaver platform offers a pallet of particular solutions for the integrated applications that enable the common work – the inter-communication among the modules, building applications based on other applications, in order to cut the costs afferent to the creation of new applications for business management.

B. The Architecture of Web Dynpro ABAP

Web Dynpro is a framework specially created to realise web applications on the SAP Netweaver integrated platform. On this platform, two application servers are integrated: Application Server ABAP and Application Server Java, enabling the programming of the applications either in ABAP (language owned by SAP) or in Java. The Web Dynpro ABAP is integrated in the ABAP Workbench and contains both an execution and a graphical development environment with special Web Dynpro tools. This framework is realised in according to the Model View Controller (MVC) paradigm Fig.2.

The MVC is a design pattern that separates an application in three components (model component, view component and controller component). In the model component, all the application data are processed and returned to the controller. The view component defines the user interface, and the controller component interprets the data and enables the link between the other two. As in [4], in case of the Web Dynpro framework, the entire structure is made of components, the separation of the data from the design part being realised through the objects that generate the data and the objects that consume the data.

Some of the advantages offered by Web Dynpo in developing the web applications, presented in [6] and [7], are:

- Possibility to use graphical tools, large offer of technologies, e.g. HTTP, HTML, CSS, XML and client-side scripts that are the base of any Web application.
- Strictly separation between the data presentation and processing.
- Possibility to use and reuse the components.
- Easy modification of the application due to the tools it disposes of.
- Possibility to access the data from the application context that remain intact even when the page is changed.
- Minimise coding, maximizing design.
- Automatic data transport through data binding.
- Automatic check of the inputs.
- Access to the user interface.
- Completely integration into the ABAP development environment.

In comparison with WD Java, which is heavily used by System Integrators to create new processes using open...
software solutions and services, WD ABAP is widely used by the SAP Business Suite to create the mainstream Business Suite applications.

C. Implementation of a Web Dynpro application on integrated system SAP

Generally, a web application displays an interface of which design is achieved by UI elements (buttons, labels, links, tables, etc) and makes the connection with other interfaces. Such interfaces can be easily achieved on the SAP integrated system by the Web Dynpro ABAP framework. Because the main architecture of the Web Dynpro ABAP framework is based on the Web Dynpro Component, a web application is consisting of such component. The structure of such component, based on the MVC principle, has as base elements ComponentController, Component Interface, Views, Windows and Web Dynpro Applications.

The interface with other components, if these shall be defined as usage, is achieved by Component Interface which is integrated part of the web component. The connection between the data model and the presentation part, as well as the connection with other components or data models is achieved by Component Controller.

Compared to other PHP frameworks (i.e. Prado, Picora, Qcodo) that need more applications for achieving a web page (text editor, graphic editor, browser, Internet connection), the SAP system provides a great advantage due to the multitude of integrated users, by means of which could be built complex web applications. Such applications are achieved on the system by SE80 transaction (Object Navigator), by means of the Web Dynpro framework. Due to the Model View Controller principle, on which is based this framework, and the usage mechanism, the applications could be achieved with more techniques as in [8]. One of the most used, due to its efficiency, is the faceless component-making technique, that connects the logical part to the design through Controller, structuring the entire application by components. Each consisting part from a component (the Component Controller, theView(s), the Window) disposes of own elements (i.e. Properties, Context, Methods) that are established depending on the application’s specific. The components built as such could be reused in other applications.

The components that are built to achieve the user interface are integrated into a component aimed for this, and the interfaces’ design is achieved in the Views’ Layout by means of the UI(User Interface) elements from the framework’s library. These elements are easily added, by drag and drop Fig.4 and have a large range of properties, as compared with the Prado framework, in which must be written a code for creating the UI element in page file, and for implementing an action for a Button-type element, it should be achieved into a file with php classes.

Population with data of the UI elements is achieved by binding them to a context that can be achieved by nodes and attributes resulted from tables. These tables can be created and related directly on the system by means of the DDIC utility (Data Dictionary) integrated in the system’s ABAP Workbench environment. Thus, the data are binded automatically by the program code by data binding. In DDIC can also be created objects as data types, type groups, domains, search helps, views and lock objects.

The component built for achieving logical parts of the entire web application, shall be a component without grafic elements, formed only by Component Controller, which, by its structure, will contain exactly the nodes of the components that forms the specialised component from the presentation.

The methods applied to the actions that don’t require navigation are transferred in the controller of the Model component, as interface methods, to detach the model from the visual part. The beauty of the Web Dynpro programming model is that the relations among different Controllers are declaratively created. It’s no need to write a code to be able to transfer data from one controller to another one. The transfer is realised through internal or external mapping, and the only code required to be written is the one which populates the component controller context. This population is simply realised by using the Supply Function method that refers directly to the node and its elements and it’s partly generated by the framework. The applications server makes automatically the connection to the database and, by Open SQL instructions, the data could be easily processed.

An important role in achieving the code have also the interfaces existant on the system. These interfaces contain all the methods necessary for programming the application’s events. All the data manipulated in the application are found in the contexts that define nodes of Input Element and Interface Node types.

![Figure 4. View Layout overview](image-url)
Besides interfaces, the Web Dynpro framework offers implicate methods (Hook) and specific methods for the ComponentController and for the View or Window.

They can be used to initiate the controller (WDDOINIT), to execute its closing (WDDOEXIT), or to dynamically program certain elements from the View. Just as in the Prado and QCodo, for the effective programming there is one more Web Dynpro Code Wizard that helps the programmer, by generating codes for certain options. For example, the programming of the ONACTIONENTER_BACK method is partly realised with the Code Wizard used for reading the node from the context, and for navigation Fig. 5. The remaining code is compiled according to [9] and [10] ABAP programming.

![Figure 5. Code generation by using Web Dynpro Code Wizard](image)

The component that interprets the data and makes possible the application’s presentation is the “Controller” component. According to the usage mechanism, are added the “model” and “view” components, and because all data are vehicled through the nodes in the context, we shall chart internally and externally these nodes by means of the Interfacecontroller.

On this structure will be created the web application’s component, to which the framework will attach a link, that can be tested in the browser. Compared to the Prado framework, in order to create an application, we need an index.php file that will be displayed at the user, and other certain components, among which a page and a file with php classes. The file Index.php shall have the following content:

```php
$application->run(); // run application

require_once('path/to/prado.php'); // prado script

$application = new TApplication;
require_once('path/to/prado.php'); // prado script

$application->run(); // run application
```

Of great help to make code, in the system exist interfaces which have a lot of methods. These methods are used in different programming requirements.

The Controller component, which links the components View and Model, is the basis of the web application. So, this component will define the usage for the two components (View and Model), will in the context of ComponentController internally map the nodes included in the Model and will instantiate these components in the wddoinit( ) method.

II. CONCLUSION

In this work we performed a study regarding the web programming on SAP integrated system, study materialized by achieving of a business web application and made by means of the Web Dynpro technology and the faceless component-making technique.

Web Dynpo is one of the top technologies used to realise high quality web applications, and the componentisation is the technique used to build web projects on the nowadays SAP platform. The componentisation, as a technique based on the Model View Controller design pattern, enables the individual components of a project to be independent from the other ones, to be easily changed and extended. Anytime the application needs modifications, these can be easily made without affecting the application’s structure.

The detachment of the data model from the visual part, although requires a greater programming effort, brings a maximum win, because these components can be reused, the entire project can be extended and easily managed and, the last but not the least, it offers the possibility to use in common the data models.

Thus, a web application, implemented on a SAP integrated system, is much more easier to achieve compared with its achievement by other Web technologies. And that because the Web Dynpro framework is one of the most actual and complex frameworks, both its architecture and the web components, based on which the applications are achieved, operate on the same MVC principle.

In present, based on Web Dynpro technology, SAP Business Suite enhancement packages from the SAP NetWeaver platform, deliver the development of new and attractive application user interfaces (UIs). Web Dynpro demonstrating the flexibility of SAP’s UI strategy for enterprise applications, through integration with other Web technologies (e.g. Floorplan Manager, POWER List).

REFERENCES