

Improving Business Intelligence Applications by Using New Generation of Web and Mobile Technologies

Mihaela-Laura IVAN

Bucharest University of Economic Studies, Bucharest, Romania

ivanmihaela88@gmail.com

The current paper presents an overview about the combination of the new technologies like: SAP HANA, SAP UI5 and SAP Fiori. There are presented the many advantages of which the final user can benefit, such as: fast performance when querying a request, data modelled with the last database innovation of SAP, SAP HANA database, which brings the biggest feature which is in-memory processing. This in-memory computing of SAP HANA enables people to focus on innovation. In the third section are described two examples of web and mobile applications developed with these technologies.

Keywords: SAP HANA, SAP UI5, SAP Fiori applications, Performance, In-Memory

1 Introduction - Fiori Apps with SAPUI5

SAP Fiori comes with modern design for a rediscovered user experience. SAP Fiori user experience (UX) represents a personalized user experience for multi-devices, responsiveness interface and with deployment options. When combined with the power of the SAP HANA platform, SAP Fiori provides exclusive responsiveness. SAP Fiori UX is used for enterprise apps across the most business scenarios. It provides performant usability on multiple devices for the best business correlations and ease of use. Now users can be happy with these real-time apps delivered with the help of Fiori apps. Below are described specific characteristics of Fiori apps [1], [2].

- Fiori apps are in fact Web apps rendered via browsers like: IE, Safari, Google Chrome;
- Multi-device enablement: same app can be launched from the desktop and any mobile device [3];
- Responsive design: rendering of an app adapts to the device (desktop or mobile device) from which it is launched;
- Role-based: user launches the Fiori apps from a home page in which all apps assigned to the user's role are bundled;
- SAP UI5 in Blue Crystal design.

2 Modelling in SAP HANA

SAP HANA ambition is to deliver across all five dimensions: deep, high-speed, simple, real-time and broad. Also SAP HANA platform makes real to combine OLTP and OLAP patterns in one application. This was possible because of the evolution of hardware innovations. In detail, these innovations are: multicore architecture (8 CPUs x (8-16) cores per blade) and address space (64-bit address space) [4], [9].

The features of Modelling in SAP HANA are as follows [5], [6]:

- The types of Joins available in SAP HANA are: Referential, Inner, Left Outer, Right Outer and Text Join;
- Information Models (SAP HANA Views) are optimized for SAP HANA Engines and Calculation Operators;
- Columns can be Attributes or Measures in SAP HANA Views. Attributes are descriptive data, while Measures are key figures with measured data;
- Using for modelling the specific views: Attribute View -> Analytical View -> Calculation View.

Table 1 below presents the strongest benefits of SAP HANA Views. It is realized a comparative analysis of the *Attribute View*, *Analytical View* and *Calculation View* [7], [8].

Table 1. Benefits of SAP HANA Views

Attribute Views	Analytical Views	Calculation Views
<ul style="list-style-type: none"> - Description Mapping & Text-Join Lookup master data tables e.g. used for handling of multi-language master data - Various types of joins of tables in data foundation or using existing attribute views - Time Dimension Attribute Views are supported – two types of Hierarchies: Level Based - it is one to one mapping and Parent-Child - it is one to many - Calculated Attributes (static or dynamic calculations) - Define filters on columns - Attributes Views can be used as dimensions in Analytical Views 	<ul style="list-style-type: none"> - Use Logical Join and Data Foundation - Data Foundation: the tables used for making the fact table of the view - Logical Join: the relationship between the fact tables and attribute views - Design time filters and input parameters to limit the dataset - Use Input parameters to calculate measures based on user input - Assign variables to further restrict the dataset for better performance 	<ul style="list-style-type: none"> - Calculation Views are usually build upon Analytical Views, Attribute Views, and Column Tables. - SAP HANA Calculation Views provide the means to model sophisticate views based on normalized data structures. - SAP HANA Calculation Views typically feed data to Business Applications, like SAP HANA XS developed Applications - Complex Calculation Views demand a more explicit intent and control of the modeled set-based data flow, examples: aggregation and filtering of sets as input to unions, joins. - After the view creation was done, we can directly analyze the data in different perspectives with the help of 'Data Preview' feature. The three options available in Data Preview are: Raw Data, Distinct Values, Values.

As displayed in Table 1, each view has its own characteristics and each of them can be used for Modelling in SAP HANA.

3 Applications Using the Future Innovation of Technologies

To emphasize what can be done with the mentioned above technologies, I present in this paragraph some examples of applications which use the latest and the best technologies. The first example of application developed with the mentioned technologies is

‘Measurement Objects Repository application’ which brings the following main features:

- Interfaces to Master data: Product Taxonomy, Solution Taxonomy and Product Feature;
- Measurement Object assignment to multiple reporting structure (Product / Solution / Product Feature and Product Owner defined);

In order to illustrate the user interface of the app, I have inserted below some relevant print screens.

Firstly in Figure 3, the User Interface of the final Fiori-like SAPUI5 application described looks like below. When pressing the *Add*

button a view named ‘*Add Measurement Object*’ is opened on the right. After filling the fields the user may add the newly created measurement object to the table containing measurement objects. This screen is of laptop (desktop) view version.

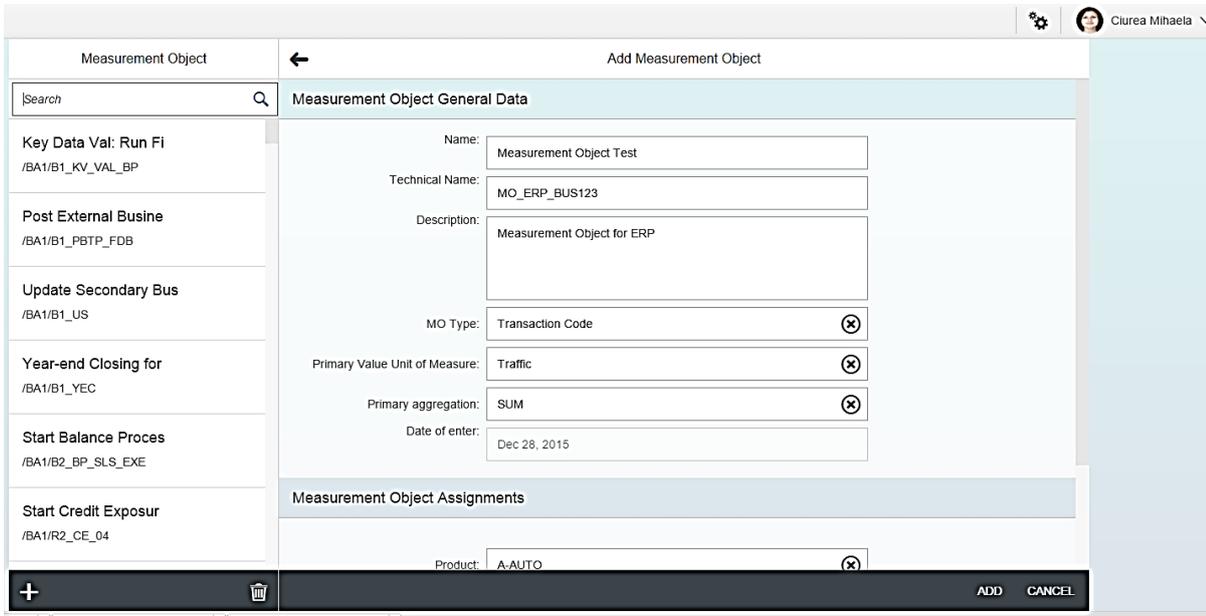


Fig. 3. Adding Measurement Object in desktop view version

Data displayed on the left part of Figure 3 comes from an attribute view as it is shown in Figure 4.

It's better to use an attribute view than directly a table because the first one is much

performant. This happens because of Join Engine used when querying this attribute view.

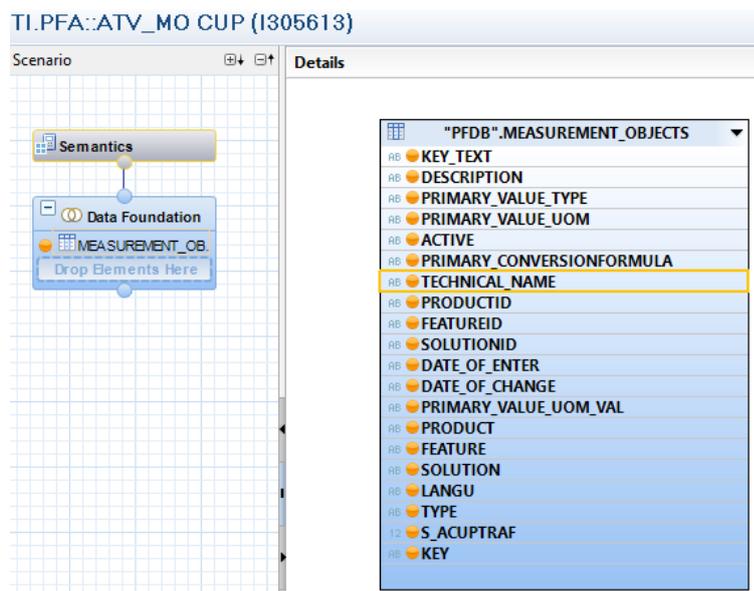


Fig. 4. SAP HANA Attribute view with one table

Secondly in the following caption (Figure 5) is represented by mobile version of the application. As can be seen, the application is responsive on the selected device and appears a collapsible view only in mobile version.

The user tries to search the newly created measurement object 'Measurement Object Test' with technical name 'MO_ERP_BUS123'. The user is able to search by name and technical name. The search is performant because of fuzzy search feature of SAP HANA which helps speed up the search capabilities when having large amounts of text. Fuzzy search functionality enables finding strings that match a pattern approximately. It is a type of search that find matches also when users misspell words or enter only partial words for the search [6], [10].

From user interface perspective it is used a `sap.m.SearchField` SAP UI5 control.

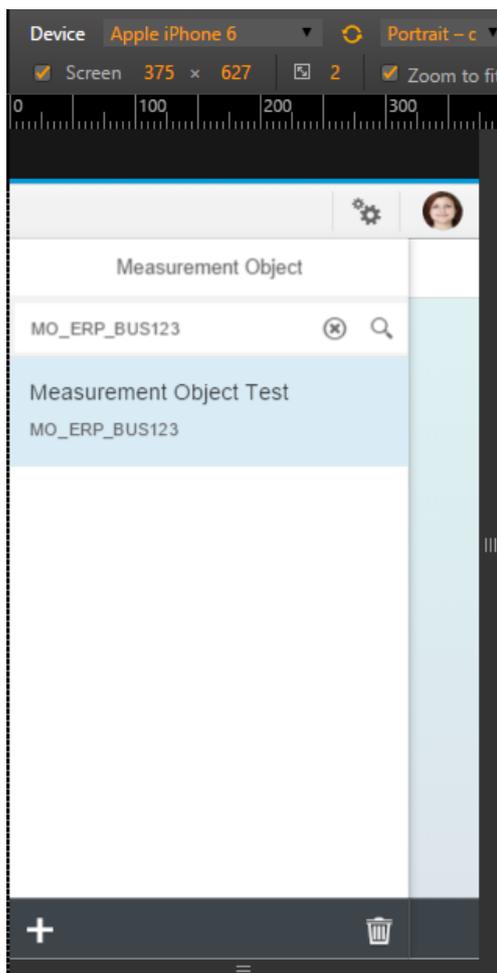


Fig. 5. Searching Measurement Object in mobile view version

In the Figure 6 below is presented another important app's feature, namely the 'Assign Measurement Objects to a package'. As a flow, the user can search the measurement object desired; if that MO exists then the user can check it and add to the current package the user is in, otherwise the user can choose to create a new Measurement Object, directly from this screen. Also, this view has measurement objects not yet assigned to the current selected package.

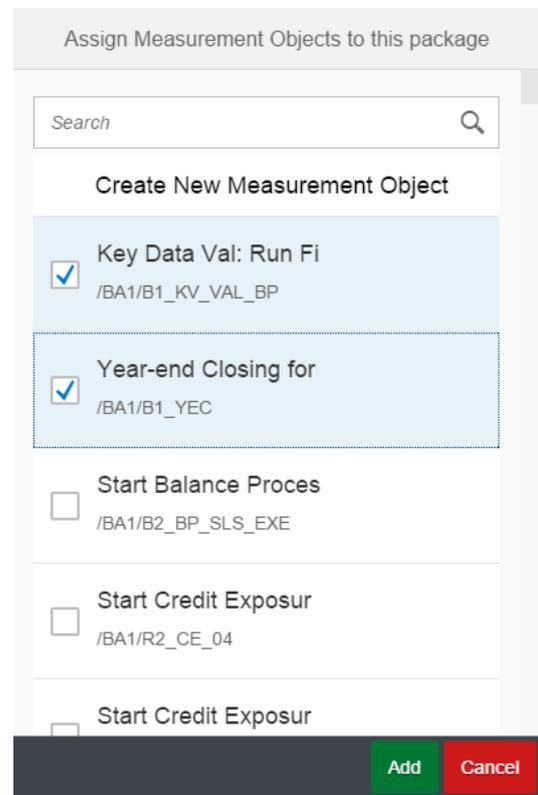


Fig. 6. Adding existing Measurement objects to a Package

Data for the above screen comes from an SAP HANA Calculation View of type standard and looks like below (Figure 7).

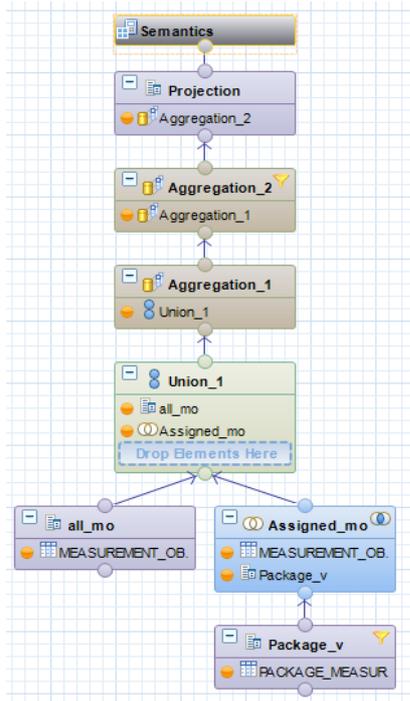


Fig. 7. SAP HANA Calculation View

As it can be seen in the figure above, a Calculation View can contain Joins, Unions,

Projections and Aggregations. The scope of this Calculation View is to display data not yet assigned to the current package selected.

That’s why the logic behind this view is: all measurement objects – assigned measurement objects = not assigned measurement objects. Also from SAP UI5 perspective, in figure ‘Adding existing Measurement objects to a Package’ are some UI5 controls and explicitly these are:

- A `sap.m.Dialog` control which is in fact exactly this screen;
- Then this `sap.m.Dialog` has in its content other three controls: `sap.m.SearchField`, `sap.m.ActionListItem` with the text ‘Create New Measurement Object’ and `sap.m.List`.

The Figure 8 represents the traffic per measurement object, in this case it’s ‘No. of Enhancements: BAdI Implementations’ measurement object with primary value unit of measure: traffic and the traffic’s value is 437319.



Fig. 8. The Measurement Object’s last month’s usage

This graphic is based on a calculation view of type script created with SAP HANA in SAP HANA Studio tool. Below it’s the procedure

script of this view which has columns: CALMONTH, USAGE, and Input Parameters: MO, PPMS_ID.

```

/***** Begin Procedure Script
*****/
BEGIN

    declare ppms_id1 string;
    select productid_associated into
ppms_id1 from pfdb.packagemo where key =
:PPMS_ID;
    var_out = SELECT "CALMONTH",
SUM("S_ACUPTRAF") AS "USAGE" FROM
"HANADB"."/A1SSPC/CRP_D_AOUE00400"
WHERE "S_ACUPMO"=:MO AND
"CALMONTH" >
(TO_NVARCHAR(TO_DATE(ADD_MONTHS(CURRENT_
DATE,-12)), 'YYYY') ||
TO_NVARCHAR(TO_DATE(ADD_MONTHS(CURRENT_D
ATE,-12)), 'MM')) AND "S_ACRM_SRO"='PROD'
GROUP BY "CALMONTH"
ORDER BY "CALMONTH";

END
/***** End Procedure Script
*****/

```

Another application which uses these combination of technologies is SAP Electronic Healthcare System (EHCS). ‘Empowering doctors and patients with mobile apps can dramatically improve patients’ healthcare, whether they are in hospital or in the home, says SAP’ [11]. To highlight how a SAP Mobile Health app can improve patient care I present below the main features of this application [12], [14]:

- Ability to add personal info, dependents’ information, Policy information. Name, Date of Birth, Gender, blood group, Address, Height & Weight setting, contact information, policy dates etc. If the user enters for the first time, then this page is displayed else Home page.
- Photo upload for end user/dependents
- The Home page header have photo, name, blood group, number of dependents, Angels, Notifications
- On click of dependents, cards are displayed below with carousel. Display dependant information upon selection of a dependent.
- Ability to maintain information for dependents
- On click of Angels, cards are displayed below with carousel for end users’ Angels and users for whom the end user is an Angel. Display information of user for

whom the end user is Angel upon selection. The end user can add/remove Angels.

- Notifications in Home page with reminders, system generated notifications, alerts.
- In the MyInfo/overview tab, tiles for monitoring various parameters are shown. E.g.: Weight, Height, BMI, blood glucose etc. All information in the tile have chart and tabular display support.
- The end user is able to add allergies and maintain the same. System prompts a list. The end user can also add specific ones.
- Integration with Nike+ and Fitbit. The information from these entities would be displayed in a tile. User is able to login inside the tile to authenticate and get information. These tiles would give information on food intake and activity information.
- The end user must be able to add a tile by choosing a parameter to monitor.
- In the medications tab, ability to add details of symptoms, diagnosis, doctor, hospital, treatment information, upload diagnostic reports, images of X rays, scans and videos. The user can also edit/delete information at any point in time. The uploaded content is shown as an overlay inside the application whenever the user wants to view it. The data is displayed in timeline format.
- The information maintained in the medications tab is searchable.
- The end user is able to add details of medications to the end user as well.
- In the emergency tab, details of contacts, policy information and map with hospitals for the end users current location along with traffic information.
- In the map section, the end user can search places. E.g.: pharmacy in Bangalore. The search results are sorted by 'time to reach' and not by 'distance'. Information is shown with ratings, reviews, address, contact information, open/closed.
- Additional options in Maps like ATM, Police station, Transit.

- The end user can drag and move around the tiles in My Info/Overview. This needs to be retained if the user enters the application again.
- Pre-emptive notifications according to age and gender suggesting the user on the importance of certain medical examinations. e.g.: Vaccines or routine health check-up suggestions for babies and elders based on the age and health conditions of family members. For example, “Your son Adam is turning three next week. Please plan for Hepatitis vaccination”. The user can convert these notifications into Reminders.

4 Conclusions

The future of mobile applications sounds good when combining SAP HANA and SAP UI5 to construct SAP Fiori apps because the user can benefit of the following advantages: intuitive UI of managing objects, accessibility via mobile, fast performance with HANA and minimum data maintenance, no redundancy of data maintenance [13]. Reporting in HANA means to use modelling views: attribute views, analytic views and calculation views and to see the results in HANA Studio tool or Eclipse tool. Then SAP Fiori consumes this views by taking the data from these views and show in a nice, easy to understand format. In conclusion the final user can benefit of simple, fast, coherent, role-based and responsive Fiori apps.

References

- [1] T. Marz, F. Berg, B. Ganz, O. Graeff, *Building SAP Fiori like UIs with SAPUI5 in 10 Exercises*, SAP AG DJ Adams, Bluefin Solutions, January 2014.
- [2] B. Ganz, *Get to Know the UI Development Toolkit for HTML5 (aka SAPUI5)*, SCN blog, September 2012, Available at: <http://scn.sap.com/docs/DOC-31625>
- [3] SAP AG, *UI development toolkit for HTML5 - Demo Kit (Documentation, API-Reference)*, Available at: <https://SAPui5.HANA.ondemand.com>
- [4] S. Patnaik, *HANA Modeling Good Practices*, SCN blog, March 2014, Available at: <http://scn.SAP.com/community/services/blog/2014/03/26/HANA-modeling-good-practices>
- [5] *SAP HANA Introduction*, HA100, 2013, SAP AG
- [6] *SAP HANA Implementation and Modeling*, HA300, 2013, SAP AG
- [7] *SAP HANA Developer Guide*, SAP HANA Platform, SPS 08, Document Version: 1.1, 2014-08-21.
- [8] V. Nandimandalam, *Attribute View creation in SAP HANA (Using Single and Multiple Tables)*, March 2012, Available at: <http://scn.SAP.com/docs/DOC-26093>
- [9] *SAP HANA Modeling Overview*, November, 2013, Available at: https://hcp.SAP.com/content/dam/website/SAPHANA/en_us/Technology%20Documents/DOC-1216%20SAP%20HANA%20Modeling.pdf
- [10] *SAP HANA Tutorial*, 2015, Available at: <http://SAPHANAtutorial.com/>
- [11] E. Lai, *How Two SAP Mobile Health Apps Can Improve Patient Care (mHealth)*, ÜberTech, September 2011, Available at: <http://www.zdnet.com/article/how-two-SAP-mobile-health-apps-can-improve-patient-care-mhealth>
- [12] M. L. Ivan, M. Velicanu, “Healthcare Industry Improvement with Business Intelligence,” *Informatica Economică*, Vol. 19, No. 2, 2015, pp. 81-89.
- [13] M. L. Ivan, “Characteristics of In-Memory Business Intelligence,” *Informatica Economică*, Vol. 18, No. 3, 2014, pp. 17-25.
- [14] M. L. Ivan, M. Velicanu, I. Taranu, “Using Business Intelligence in Healthcare System,” *The 14th International Conference on Informatics in Economy, IE 2015*, 30 April-03 May, 2015, Bucharest, Romania, ISSN 2284-7472.



Mihaela Laura IVAN graduated from the Faculty of Cybernetics, Statistics and Economic Informatics of the Bucharest University of Economic Studies in 2011. She also finished the Master's degree in Economic Informatics in 2013, at the Bucharest University of Economic Studies. Starting with 2013, Mihaela is a PhD candidate at Bucharest University of Economic Studies in the field of Economic Informatics. At the present, she is a SAP Development Consultant at SAP Near Shore Centre Romania.