Overview of the use cases provided

With the SAP IBP, add-in for Microsoft Excel (Excel add-in) version 2202.2.0 and newer, we provide further APIs to make basic functions of the Excel add-in accessible through Visual Basic for Applications (VBA) code. You can use the APIs to enhance your planning view templates through VBA code to build your own business logic, and automate some of the manual steps planners need to do in their daily work.

As an example, you might build interactive templates with customized drill-down or navigation possibilities. Creating customized templates for each business process or user group improves the user experience and makes it easier to fulfil the planning tasks which need to be done. We encourage you to build your own use cases and enhance the Excel add-in functions to suit the needs of your business users.

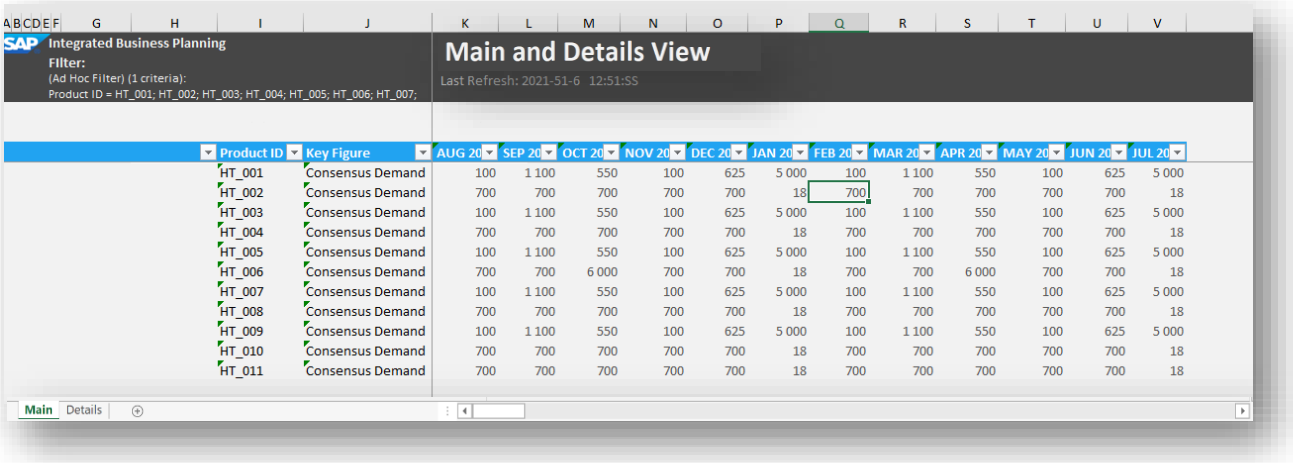
In this document, we explain the different business use cases we were thinking of when implementing the APIs. You can use these as a starting point to collect ideas and to explore possible use cases of the APIs.

1. USE CASES

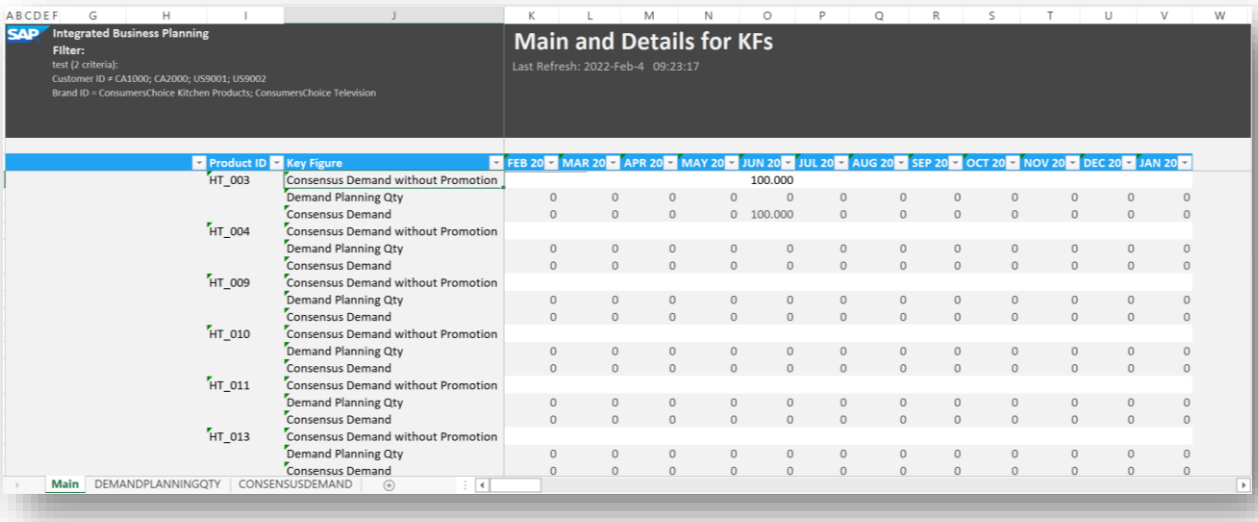
In the following subchapters you will get a high-level overview of the different uses cases and how they can be used. For each use case, you will find a specific folder with a detailed description and guidance for setting them, including ideas for further enhancements, along with an .xlsm file with the respective code samples.

1.1. Create customized navigation patterns

In this use case, we explain how navigation patterns between worksheets in a planning view template can be set up.
For example, navigating from a main planning view worksheet to a more detailed planning view worksheet by double-clicking a cell (as shown in the screenshot below). Such a planning view template can be used for a review meeting where the key figure values of different combinations (in this example only Product ID is selected, but it could be a combination of Product ID and Location ID as well) are reviewed and checked. Whenever further details are needed for a specific value, you can navigate to the details worksheet by double-clicking the cell. The attribute values of this cell are passed as filter settings and the details planning view is opened for this specific combination (in this case the specific Product ID).

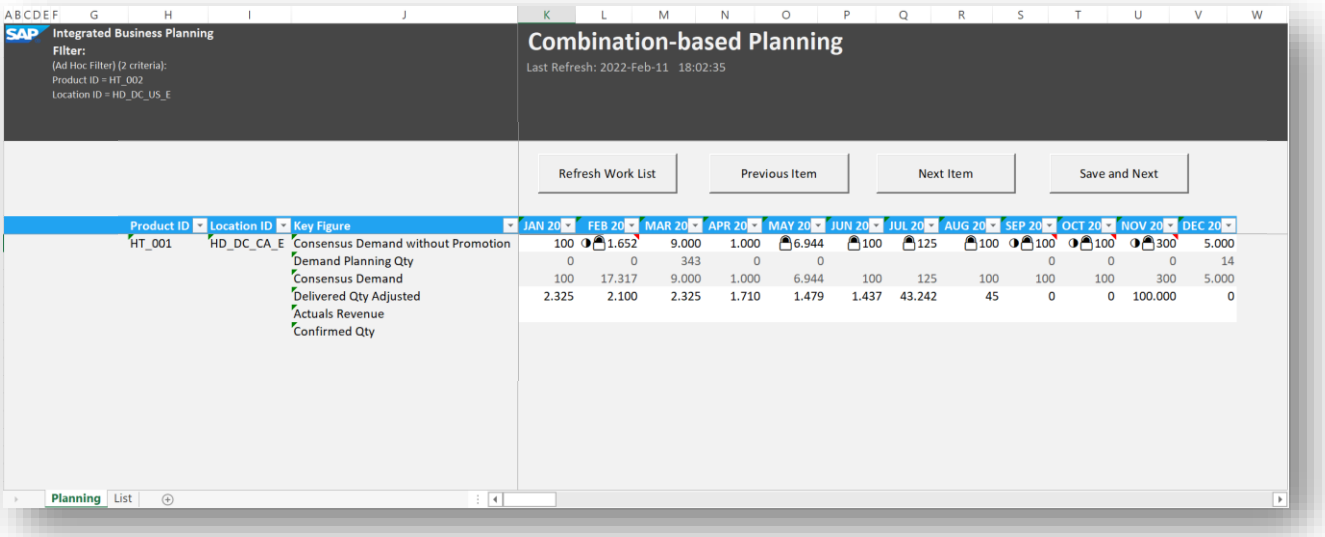


The second example we are providing for this use case is to navigate from a main planning view worksheet to different key figure-specific worksheets by double-clicking the key figure (see screenshot below). Such a planning view template might help the planners in their daily work. If further details are needed for a specific key figure and combination (in this example Product ID), they can double-click the key figure cell and easily navigate to the details sheet. As in the example above, the attribute values of the key figure cell are passed as filter settings and the detailed planning view is shown for this specific combination.



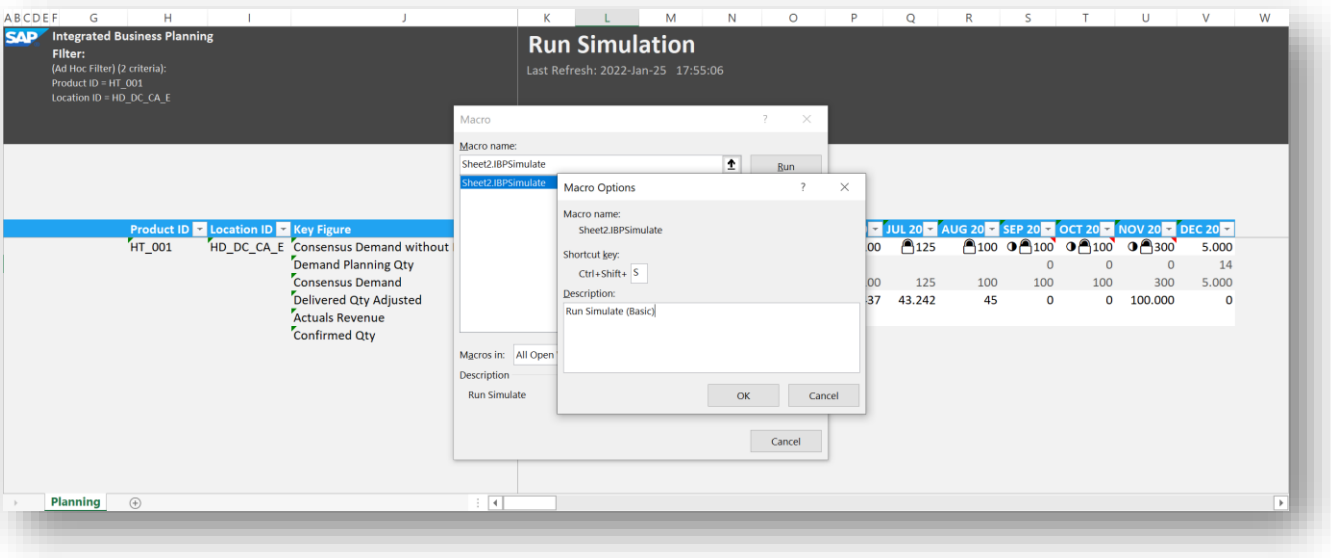
1.2. Combination-based planning

Configure a planning view template where the planners can go through the combinations that they are in charge of one by one, entering their changes, and clicking the button “Save and Next”. If no changes are needed or if the previous combination needs to be checked again, the buttons “Previous Item” and “Next Item” can be used. By clicking “Refresh Worklist” the worksheet containing the list of combinations is refreshed in the background, and the attribute values of the first combination are used as filter criteria for the “planning” worksheet.



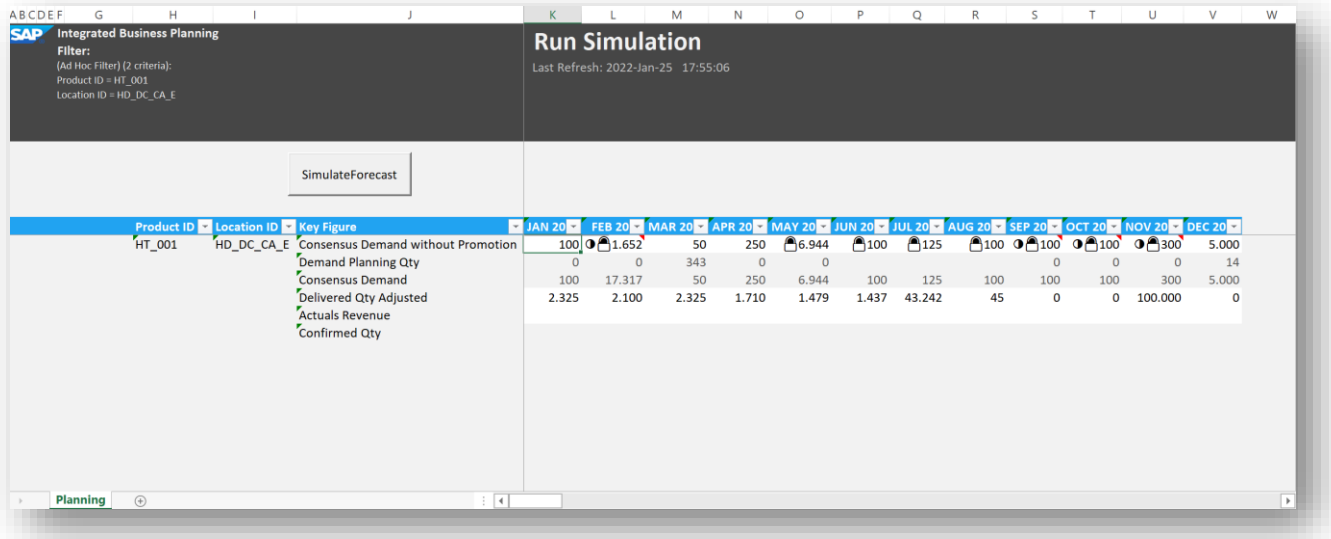
1.3. Trigger SAP IBP actions using shortcut keys

In Microsoft Excel it is possible to assign macros to shortcut keys. Using this function, you can assign actions of the SAP IBP, add-in for Microsoft Excel (Excel add-in) such as triggering Refresh, Saving Data or Simulate (Basic) to specific shortcut keys (see screenshot below).
You can also assign more complex macros to shortcut keys, such as the logic of the button “Save and Next” from the sample “Combination-Based Planning”.



1.4. Run simulations using VBA buttons

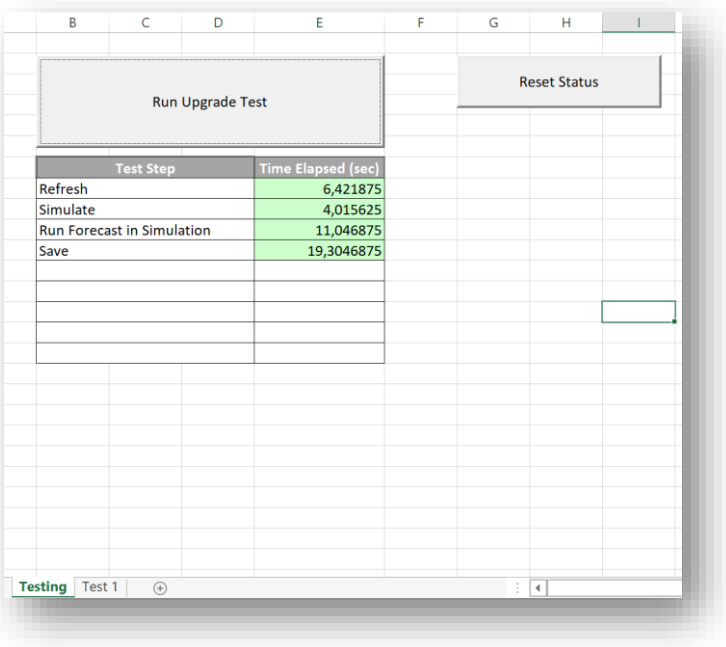
Create buttons for running simulations for specific planning operators.
If a user group always uses the same statistical forecast model in simulation mode, you can include a VBA button into the templates for running the specific operator in simulation mode.



1.5. Measure the performance of SAP IBP actions

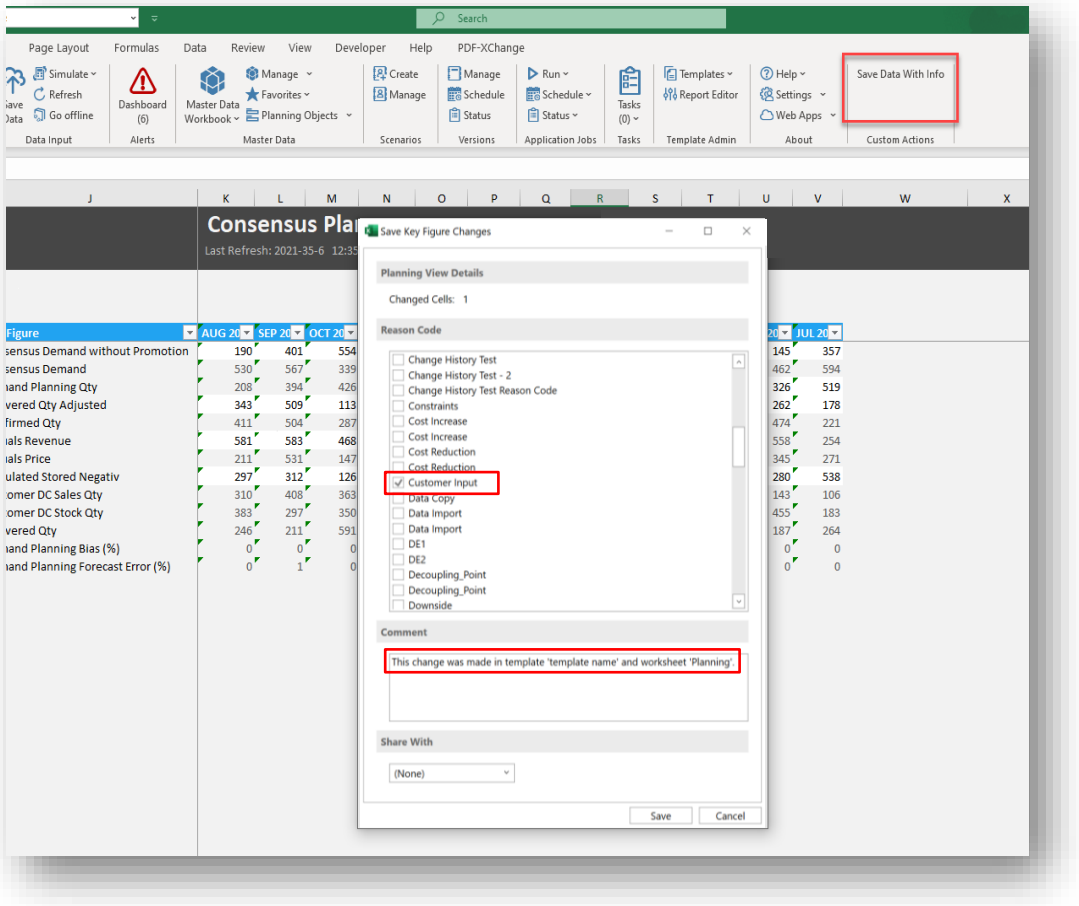
One advantage of using VBA code is that several steps can be combined. Within one button, different simulation runs can be triggered, and data changes can be made and saved.

In our example, we explain how you can set up an upgrade test to measure the performance of different actions in the Excel add-in after an upgrade of your SAP IBP system.



1.6. Prefill reason codes and comments when saving data

Trigger Save Data with prefilling reason codes and comments and optionally suppressing the dialog for reason codes, comments and sharing.
Additionally, the SAP IBP ribbon can be extended by using an .xml file and VBA code as shown in the screenshot below.



Follow us

www.sap.com/contactsap

THE BEST RUN