Clean Core: A paradigm shift for SAP applications

In order to understand the topic of Clean Core, it is worth taking a look at the past. In the vast majority of cases, SAP systems were installed as an on-premise version on in-house servers or those of a service provider. The functional scope of the individual SAP modules was not always at the same level, meaning that customerspecific requirements could not always be fully implemented with the standard.



Clean Core: A paradigm shift for SAP applications

In order to understand the topic of Clean Core, it is worth taking a look at the past. In the vast majority of cases, SAP systems were installed as an on-premise version on in-house servers or those of a service provider. The functional scope of the individual SAP modules was not always at the same level, meaning that customer-specific requirements could not always be fully implemented with the standard.

We saw these problems in the SAP LE-WM and SAP LE-TRA modules and implemented adjustments via user exits and BAdIs (Business Add-Ins) directly in the application code.

To differentiate the code from the SAP standard, enhancements were and are implemented in a separate namespace at Körber. Unlike enhancements that are implemented as modifications, the application therefore remains releasable.



Public or private cloud?

In recent years, SAP modules have been enhanced or completely redeveloped by SAP. Examples of new developments include SAP EWM (Extended Warehouse Management) and SAP TM (Transportation Management). The technical scope has now matured to such an extent that customers can decide whether they want to introduce a genuine standard product that can only be customized to a limited extent or whether they want to retain flexibility and introduce a customer-specific, private system.

In the former case, deployment takes place in the SAP Public Cloud. Even if most people are aware of the difference, it should be mentioned again at this point that a public cloud application is provided simultaneously for several companies on one instance. Data is separated at database level. Mutual access to data is therefore excluded. The "private" application is provided in the SAP Private Cloud.

What does this have to do with Clean Core?

A key feature of a standard application is that it is continuously developed further. This also applies to SAP applications. New functions, technical improvements or bug fixes are made available in new patches and releases. The higher the proportion of individual adaptations in an application, the greater the effort required to install the updates.

In an ideal world, a standard application would manage without customizations. The core of the application is clean or free of customizations. Installing updates in a clean-core application is not a problem here. This is the guiding principle of a public cloud application. The customer is provided with an application that they cannot customize or can only customize to a very limited extent. Maintenance of the application, including the installation of updates, is carried out by SAP. The testing effort for the customer is supposedly low.

For applications that serve a complex topic such as SAP EWM for automated warehouses the clean core approach is currently a challenge.

What does this mean in technical terms?

The following illustration shows four variants of how standard applications provided in the SAP cloud can be extended.



These 4 variants are available:

- Variante 1: Key User (In-App) Extensibility in the SAP S/4HANA Core
- Variante 2: On-Stack Developer Extensibility
- Variante 3: Key Side-by-Side Extensibility on the SAP Business Technology Platform
- Variante 4: Classic Extensibility

Key-User (In-App) Extensibility

Key-User (In-App) Extensibility enables the customization of user interfaces, processes, e-mail templates or forms according to the low-code or no-code principle. Consultants as well as key users can implement extensions with this variant. In terms of qualifications, sound knowledge of business processes and configuration is a prerequisite. Programming skills are only required if user-defined business objects or business logic are to be added using the Cloud ABAP Web Editor.

Application examples:

- UI customization for layouts such as moving/hiding fields and field groups, changing labels, etc.
- Ul customization for custom forms and templates.
- CDS views and analytical applications.
- Creation of business objects.
- Extension of fields for standard business objects. Custom fields are available from the user interface to the database tables.
- Custom business logic with Cloud BADIs .
- Add custom fields to process groups (e.g. from quote and order to delivery and invoice).
- Copy and customize print and email form templates.

On-Stack Developer Extensibility

This option is located between the key user and side-by-side extensibility. The on-stack developer extension enables the programming of extensions via individual ABAP code as well as developments that require a link to SAP S/4HANA data, transactions or applications. This option is typically used when customer-specific developments are to be carried out that require full access to development functions such as debugging or version control.

Application examples:

- ABAP-based custom applications and extensions developed with a new cloudenabled ABAP RAP (RESTful Application Programming) model on shared APIs.
- Custom applications with SQL access to SAP S/4HANA data cannot be implemented using side-by-side or data replication.
- Custom extensions that run in the same Logical Unit of Work (LUW) as SAP applications.
- Custom remote APIs or services for side-byside SAP BTP applications.
- Extension of the SAP Fiori application.

Side-by-side Extensibility

This option is the preferred variant for the development of extensions that take place against the background of process automation or relate to SaaS applications that are to be loosely coupled but seamlessly integrated. Extensions are developed separately on the SAP Business Technology Platform (SAP BTP).

Application examples:

- Proxy applications for a separate target group (no ERP users).
- Convenient applications that should run separately from ERP.
- Individual application that runs in parallel to ERP to reduce the load on the operational system.
- Integration of partner solutions.
- ABAP and non-ABAP developments (Java, Node.js, etc.).
- Extension of the UI application with a no-code application such as SAP Build Apps.

Classic Extensibility

It is important to know that customizations in the SAP Private Cloud are also possible using the traditional approach via user exits and BAdls, among others. A user exit makes it possible to extend the standard SAP code at certain points and at defined times without changing the code directly. With a BAdl, source code units can be extended without modification.



Advantages of Clean Core

It has already been mentioned that Clean Core simplifies the import of updates. A few associated advantages are:

- Shorter release cycles (due to the high effort involved in grown environments, updates are delayed or not imported at all).
- Standardized business processes.
- Possibility of customization by key users.
- Reduced effort in preparing and installing updates. (In an ideal world, testing is no longer necessary)
- Calculable costs.
- No support and maintenance fees for unneeded artifacts.

Disadvantages

- Changed approach to the design. The guiding principle is to develop extensions in such a way that they correspond to the clean-core approach.
- Increased conceptual effort.
- Increased development effort.
- Limited practicability with extensive, complex adaptations.
- Limited flexibility ("just do it", doesn't work).
- Influence on response time behavior for developments on the BTP is probably not a problem but should be checked for time-critical extensions.

Conclusion

Clean Core is an exciting approach that can certainly be seen as a paradigm shift. If you go down this path, you can very probably look forward to positive effects that can be measured in terms of time and money when installing updates. However, it is also true that a company's decision in favor of Clean Core points the way forward. Any deviation means a relapse into the "old days". Consistency and perseverance are therefore required.

The advantages of the clean core approach in terms of updatability are offset by the technical feasibility and the expected additional work involved in implementation. With regard to technical feasibility, for example, the number of standard APIs available must be checked. APIs are required for programming extensions. The number of APIs available for SAP EWM is still manageable. Furthermore, it should not be underestimated that SAP partners have to convert best practices developed in the "traditional way" over many years.

To ensure that projects do not get out of hand in terms of time, it should also be noted that individuality should be avoided as far as possible. This means that you are well advised to align the logistics processes with the functions provided by the respective applications.

As with all new things, SAP partners, customers and SAP itself will of course gain experience that will have a positive impact on methodology and development. It will probably be some time before projects can be implemented on an equal footing in terms of time and effort.

Are you interested in Clean Core and looking for support? Put your trust in Körber's in-depth SAP expertise and get in touch with us today!



Körber Supply Chain Consulting GmbH In der Hub 2-8 63110 Bad Nauheim sap.info.sc.bdn@koerber.com https://koerber-supplychain.com/