Classification of S/4 Extensibility

GREEN

- Possible in all S/4 HANA solutions (Cloud and onprem)
- These extension techniques are expected to have no impact on upgrades.

Yellow

- Allowed in SAP S/4 HANA Cloud, single-tenant edition, but not in public SAP S/4 HANA Cloud.
- These classic extensibility techniques would not prevent an upgrade, but may need regression testing afterwards.
- They may reuse standard SAP ABAP objects such as tables, data elements, function modules or classes. If those standard objects are changed or replaced as part of the upgrade, then the custom extension may no longer work as it was originally designed or may no longer compile at all.
- These risks could be mitigated if you try to use only whitelisted objects and APIs in your custom developments.

RED

- Not allowed in SAP S/4 HANA Cloud, single tenant edition
- Not possible in SAP S/4 HANA Cloud (multi-tenant)
- These extension techniques typically involve directly changing standard SAP code or some other object.
- Such changes may require confirmation or adjustment during the upgrade process, which requires customer knowledge and judgement and could not be completed automatically by SAP.

Business Logic Enhancement

Order of Preference	Technique
1	**Key User Extensibility** Use App Custom Fields and Logic to implement a BADI (using restrictive ABAP)
2	Use a **classic extensibility** technique with a stable enhancement point, that does **not require a modification key**, such as - ABAP BAdIs (SE18/SE19) - AMDP BAdIs to enhance standard SQL Script procedures - SMOD/CMOD enhancements - BTEs (Finance)
3	Use a **classic extensibility** technique with a stable enhancement point, that **requires a modification key** such - User Exits - VOFM routines such as pricing formulas
4	**Implicit enhancements or enhancement spots.** These do not require a modification key, but otherwise are much like Modifications. They enable customers to change any SAP code at the start or end of any coding block. In an upgrade, these would need to be processed in SPAU_ENH – the risk is that the enhancement point may no longer exist, or may no longer have access to the same data.
5	**Modifications** In this standard SAP code is directly changed without restriction. The customer must first obtain an SSCR Modification Key, so each change is registered with SAP. In an upgrade, these would need to be processed in SPAU – the risk is that the enhancement point may no longer exist, or may no longer have access to the same data.

Add Custom Fields to a standard SAP business object

Order of Preference	Technique
1	**Key User Extensibility** Use App Custom Fields and Logic to define additional custom fields for a given Business Context. The app can extend not only the underlying database tables, but also any associated CDS views and oData services.
2	Use **predefined Customizing Includes (CI_*)**. These are predefined Include structures of some standard SAP tables, in which custom fields may be added. The fields may be defined through customizing (hence the name), for example in Finance the CI_COBL 'coding block' structure. It may also be possible to add fields to the includes directly. All custom field names must be in the customer namespace (e.g. starting 'ZZ') to avoid any clashes with SAP fields which may be added in the future.
3	Use an **Append Structure:** Additional fields may be added to standard SAP tables modification-free using an Append Structure. These are not predefined - they may be added to any table. Fields should again be in the customer namespace (see above). Typically, such a change would be made along with associated UI and Business Logic enhancements.
4	**Extension Includes**: These may be used to add custom fields to an app's oData service, if the app cannot be changed through 'Custom Fields and Logic'. Technically these are provided as DDIC Includes in standard structures, which can be extended using an Append Structure.
X	**Direct table modification is not allowed.** However, they should never be necessary, as fields can always be added using an Append Structure.

Custom Configuration Table

Order of Preference	Technique
1	**Classic extensibility:** Define the table and generate table maintenance in the usual way (with the ability to transport the data). The table maintenance may be added to the IMG (transaction SPRO). Note that in SAP S/4 HANA Cloud, users do not have access to the IMG.
2	**Key user extensibility:** 'Custom Business Objects' app may be used to define custom tables. However, this is designed for master and transactional data - not for configuration. There is no option in this app to make the table contents transportable. However, it may be used to create custom tables for nontransportable settings, which may depend on master data that differs in
X	**Add/Change/Delete standard entries in standard table** with no table maintenance It is only allowed if it is recommended by an SAP Note. During an upgrade the custom content is likely to be overwritten and the table entries will revert to standard.

User Interface Enhancement

Order of Preference	Technique
1	**Personalization:** For some UI elements users can make changes themselves. For example, for tables, it's typically possible to change the fields displayed, the order of the columns or the overall sort order. If the requirements are user-specific, then consider any personalization options first.
2	**Configuration / Settings:** To apply changes more widely, there may be associated configuration or settings related to the app / transaction. These may control for example what fields are displayed, read only or mandatory. It's worth checking if such configuration is available ahead of using other techniques.
3	**SAP FIORI Apps:** Run Time Adaptation (RTA). Use to make simple changes like hiding or adding fields or rearranging fields.
4	**SAP FIORI Apps:** SAP Web IDE - hide controls. Suitable if you only need to hide some UI controls, but the app does not support Run Time Adaptation.
5	**SAP FIORI Apps:** SAP Web IDE - extension points. Use pre-defined extension points to change the app. These are documented in the Fiori Apps Library.
3	**Web UI Apps:** E.g. SAP S/4 HANA for Customer Management; SAP Solution Manager, SAP ChaRM SAP Screen Personas or Web UI Configuration Tool
3	**SAP GUI Transactions:** SAP Screen Personas
4	**SAP GUI Transactions:** Classic extensibility techniques such as CMOD enhancements or BAdIs. This technique may be needed to add custom fields to the UI.

Custom Development of Apps

Order of Preference	Technique
1	**Key user extensibility:** Use 'Custom Business Objects' app to implement any custom data model required. Then generate the maintenance app, code determination and validation logic as required using restricted ABAP.
2	Use 'Custom Business Objects' app as above and then use the SAP Web IDE to further enhance the UI.
3	**Classic extensibility using the BOPF framework.** This is the same application framework underlying the Custom Business Objects app. However, you may develop more sophisticated apllication by working directly in this framework with classic ABAP.
4	**Classic extensibility using the BOL / GenIL** when you develop Web UI based apps e.g. relating to SAP S/4 HANA for Customer Management; SAP Solution Manager, SAP ChaRM
5	**Web Dynpro or Floorplan Manager apps without the BOPF.** Possible if there's some barrier to using BOPF to define the business object.
6	**SAPGUI Dynpro Transactions** Generally, this is no longer recommended for new applications - it should be possible to use one of the other approaches instead.

Custom CDS Views / Programs

Type of Enhancement	Order of Preference	Technique
Custom CDS Views	1	**Key user extensibility:** Use 'Custom CDS Views' app to create a custom view using data from other published view. For example for the aggregation of quantities/values or for the definition of input parameters and filters.
	2	**Classic extensibility:** Define a CDS view using the ABAP Development Tools (e.g. in Eclipse). In this case the CDS Data Definition Language is coded directly and therefore more sophisticated selection logic may be implemented.
Programms	1	**Programs - background jobs** Classic extensibility - ABAP These are programs that should be scheduled to run regularly in the background without user intervention. Usually the programs execute some technical activities or long-running update processes.
	1	**Programs - reports** It's still technically possibleto write a report in ABAP. However, it should no longer be necessary. It is recommended to define a data source using a CDS view, then use or one of the S/4 HANA embedded analytics options.

Workflow / Emails

Type of Enhancement	Order of Preference	Technique
Workflow	1	Use Manage Workflow apps which are related to pre-defined business processes such as purchasing approval. These apps support the definition of workflows with multiple steps, recipient determination, and preconditions, for example based on value.
	2	**Classic extensibility** may be more suitable in two scenarios: 1) You may need more sophisticated workflow logic than the Manage Workflow apps support. With classic extensibility you may still reuse a standard workflow template as a starting point. 2) It may be that a suitable standard workflow exists already for the required business process, which is not yet supported in the Fiori apps. However, it could be adapted for your requirements.
Emails	1	**In-App Extensibility** using Maintain Email Templates app. Create the required email template based on one pre delivered by
	2	Prior to SAP S/4 HANA, there was no single and consistent approach to determine email texts. In some cases, it may be possible through configuration, in others an enhancement may be necessary.