



On-Premises / Privately Hosted → P21 SaaS in Microsoft Azure

Customization, Integration & Technical Readiness Guide

A successful P21 SaaS migration requires more than moving the core ERP environment. Customers should also identify, evaluate, and plan for the customizations, integrations, reports, business rules, SQL processes, and external applications that support day-to-day operations.

MindHarbor recommends beginning this assessment early so there is adequate time to determine what can migrate as-is, what requires configuration or testing, what may need redevelopment, and what is no longer needed.

Recommended First Step: Build a Customization Inventory

Create a complete inventory of the components currently supporting your P21 environment. For each item, document its business purpose, owner, criticality, current technology or data-access method, and whether it interacts with systems or resources outside of P21. (NOTE: Epicor may provide you with a “P21 Cloud Migration Summary” document with these items cataloged)

- Business rules and visual rules
- DLL extensions and DynaChange customizations
- Crystal Forms, Report Studio reports, SSRS and other custom reporting
- Custom DataStream objects and legacy InfoMaker portals
- SQL Agent jobs, alerts, functions and stored procedures
- Third-party integrations, middleware, custom web applications and external tools

To help with preparing, here are some good examples of customizations for evaluation:

Business Rule – Epicor will migrate all rules. Including exporting and importing of the visual rules. Epicor will stand up a Visual Rule Folder (template); however someone would need to handle the redeployment of the VRs (copy/move from the OLD P21 to the new P21 SaaS folder and set up the web application for them to run., this will include some configuration changes.

If already on P21 Web UI Version, the majority of all business rules and visual rules should work, excluding pre-SQL rules, or rules that are calling- interacting with outside environments (i.e. local paths, SharePoint folders, pushing – pulling/inserting data from outside of P21, etc.)

To get started: 1st step is to export a list of all your rules from P21 SQL backend, this should provide a bit more information than what is displayed in the Dyanchage Rule Organizer (export into an excel/csv doc for review) – we are happy to help with this step if needed. A developer will then review this list to better identify those rules that potentially will need some redevelopment and therefore require code review. The rule source code is needed, as well as some information about/ requirements for and description of what the rule is intended to do/objective. Alternatively, we typically recommend simply deactivating/disabling all rules in your PLAY P21 SaaS Azure instance and then turn each rule on/enable individual to test to see

if it is behaving as expected. Once you have identified any rules that are not working correctly in P21 SaaS, just let us know and we will then do the requirement and code review as noted above.

We are happy to help when ready and with whichever evaluation step you would like us to engage in (earlier in the process or once you have identified the problem rule(s)).

DLL extension – Epicor should migrate all DLL extensions (No action is needed from you)

Crystal Forms – Epicor will migrate all forms. If you encounter any issues with your form customizations (i.e. DataStream's, formats, etc.) please let us know and we will open a support ticket to address. These type of efforts tend to be resolved within a couple of hours.

Report Studio Reports – Epicor will migrate all reports. If you encounter any issues with your reports, please let us know and we will open a support ticket to address. These types of efforts tend to be resolved within a couple of hours.

Other Custom Reports (including SSRS): Custom SQL-based reporting requires additional planning because the report execution environment and data connection may change in SaaS.

- For SSRS reports that will remain in use, plan for a local or privately hosted SQL/reporting environment.
- Coordinate with Epicor regarding approved connectivity/whitelisting to the P21 SaaS Replicated Read-Only Database.
- Reconfigure report data sources to use the approved SaaS reporting database and validate performance and output.

Custom DataStream Objects – Epicor should migrate all copy from existing P21) If you encounter any issues please let us know and we will open a support ticket to address.

Portals (Legacy Infomaker Portals) – Epicor is expected to migrate existing portals; however, InfoMaker portals are legacy functionality and should be evaluated carefully.

- Determine whether each portal is still required.
- If changes or redevelopment are needed, consider whether the business requirement is better served by Report Studio or another supported reporting/application approach.

Dynachanges – Epicor will migrate / handle. Maintain an inventory of custom DynaChange items and validate business-critical behavior during SaaS testing.

SQL Agent Jobs- Require Evaluation to determine best approach/method to achieve same objective. Examples: Alerts will most likely need to be redeveloped as a schedule report. Also, jobs that call outside P21, will need an alternative approach/method (i.e. outputs to D:\SQLJob Output\ (local path))

To get started: We will need a list of your SQL Jobs, this is typically included in your Epicor “P21_Cloud_Migration_Summary”. A developer will review and provide feedback about next steps and effort.

Custom SQL Functions – Epicor should migrate – though it is our understanding Epicor might request / require review of your Stored Procedure for approval. Once approved they will add to your P21 Instance and or Stored Procedure Executor.

To get started: We will need a list of your SQL Jobs, this is typically included in your Epicor “P21_Cloud_Migration_Summary”. A developer will review and provide feedback about next steps and effort, including assistance with the Epicor Case/Ticket and supporting documentation for Epicor’s review

Integrations, Custom Web Applications, and External Tools

Every third-party integration, middleware process, external tool, and custom web application should be evaluated to determine whether its current integration method is compatible with P21 SaaS.

- Document the current method: REST API, SFTP/flat file, data upload, direct SQL, or another process.
- Identify the P21 objects/data being exchanged and the direction of each data flow.
- Document frequency, real-time requirements, dependencies, and any local/private infrastructure currently used.
- A developer can then determine what can remain unchanged, what requires configuration, and what degree of redevelopment may be needed.

To get started: we will need information about the current integration processes and methods (i.e. REST API, SFTP Flat File, Data uploads, Direct SQL, etc.), along with a list of objects that are syncing between platforms with directional. A developer will review this information to determine what changes to the integration are needed and or the degree of redevelopment needed with effort.

How MindHarbor Can Help

MindHarbor can engage at whichever point provides the most value to your team—from early migration planning and customization inventory review through technical discovery, Epicor coordination, redevelopment, testing, deployment, and post-cutover support.

- Review P21 customization and SQL inventories
- Participate in migration meetings with your team and Epicor as a technical resource
- Evaluate business rules, SQL processes, reports, integrations, and external applications
- Identify likely SaaS compatibility gaps and recommended approaches
- Redevelop or reconfigure custom solutions for the P21 SaaS Azure environment
- Assist with PLAY testing, UAT issue resolution, deployment, and post-live support

Planning Note: For highly customized P21 environments, allow sufficient discovery and redevelopment time well before the planned live cutover. Early preparation helps reduce late-stage surprises and provides adequate time for development, testing, user acceptance, and deployment.