PH7X SYSTEMS ®

0.1.0 · CHAPTER 22

Know every integration before it moves

SharePoint Migration Field Guide · Chapter 22

pH7x Systems

2026-11-06

22. Know every integration before it moves

> A migration moves one end of every integration. The other end finds out when it fails.

An integration is the part of the estate whose other end belongs to somebody else: a line-of-business system that writes into a list every night, an application that reads a library under its own identity, a script that exports a report every Monday. Most of it lives outside the content databases, and none of it complains until its first run after cutover. Chapter 2 counted the integrations and Chapter 3 weighed them as a calendar problem; this chapter decides each one, with the owner of its other end.

> **Which systems call your farm, which systems does your farm call, and what happens to each connection when the farm it knows is gone?**

The decision this chapter leads to is a disposition for every integration (keep, re-point, rebuild or retire), recorded with both of its ends, its identity at the destination, the test that proved it, and the agreement of whoever owns the other end.

Which integrations this chapter is for

Every connection with one end outside SharePoint, on a SharePoint Server 2016 or 2019 farm moving to Subscription Edition by attach (Chapter 23) or to SharePoint Online by copy (Chapters 8 and 9). There are four kinds: line-of-business data through Business Connectivity Services (BCS); applications on the client APIs (CSOM, REST, the older web services) under an identity of their own; scheduled scripts; and services the farm calls, from mail to Office Online Server. InfoPath data connections (Chapter 20) join the list; a workflow's web service calls follow Chapter 19.

What each destination still runs

WHAT THE INTEGRATION RELIES ON	SUBSCRIPTION EDITION	SHAREPOINT ONLINE
BCS: external content types, lists and columns	"No plans to retire" it "at this time"; its database is upgraded	Retired on 30 September 2024; Power Apps is the named replacement, with "no direct migration"
Credentials in Secure Store	"No current and immediate plans" to retire it; its database is upgraded and opened with the old farm's passphrase	Retired on 30 September 2025; Azure Key Vault is the named replacement, again with no direct migration
Azure ACS app-only access: a client ID and secret from `appregnew.aspx`	Unaffected: Microsoft says no action is required on premises	Stopped for new tenants on 1 November 2024 and for existing ones on 2 April 2026, with "no option to extend"
A Windows account, or a user name and password	Windows authentication on the zone it calls; Basic unsupported since 14 July 2026, and refused for new web applications since 24H2	Microsoft Entra ID; CSOM's user name and password sign-in is no longer supported
Remote event receivers, which call a service when content changes	Not retired: on-premises add-ins are outside the online retirement	Stopped on 2 April 2026 if registered through ACS; retiring on 1 July 2027 if registered with Entra ID; webhooks are the recommended replacement
Scripts on the server object model	A farm server and Windows PowerShell, not PowerShell 7; `stsadm.exe` is removed	No server: client APIs or Microsoft Graph, from elsewhere
Mail sent to a list or library	Incoming email, configured again on the new farm	"This feature isn't available": new mail stops arriving

Sources for the table: BCS retirement, Secure Store retirement, upgrade service applications, ACS retirement, retirement FAQ, deprecated or removed, what's new, CSOM guidance and email-enabled lists, all checked on 6 November 2026.

On Subscription Edition most mechanisms survive and the farm's configuration does not: "you upgrade only the content for your environment and not the configuration settings" (upgrade service applications, above). Trusts, incoming email, the outgoing mail server and web application policies that grant integration accounts access are set again on the new farm. Office Online Server, which shows Office files in the browser, has a retirement date of 1 January 2027 (lifecycle, checked on 6 November 2026); what replaces it on Subscription Edition, we do not know.

Find every integration from both ends

The farm keeps no list of who calls it; its web servers do. By default, IIS logs each request with its client address, authenticated user name, requested address and user agent (IIS log settings, checked on 6 November 2026). Filter each front end's logs for `/_api/` and `/_vti_bin/`, where client code

arrives (REST endpoints, checked on 6 November 2026), and group the requests by user name, client address and user agent. We would read at least a month, month end included: an account in those logs that nobody claims is an integration nobody has found.

Outward, the farm's configuration names most of what it calls. Each BDC model carries "which authentication protocol and credentials to use" for its external system (BCS overview, checked on 6 November 2026), and `Get-SPSecureStoreApplication` lists Secure Store's target applications. From an elevated SharePoint Management Shell on the old farm, `Export-SPBusinessDataCatalogModel` saves the models and `Get-SPTrustedSecurityTokenIssuer` lists the trusts (both checked on 6 November 2026):

```
# Reads the 2016 or 2019 farm; changes nothing. Writes one .bdcm file per model, and trusts.txt.
$ctx    = "https://<web application URL>"
$models = "<ModelName1>", "<ModelName2>"   # as the BDC service lists them
foreach ($name in $models) {
    Write-Host "Exporting BDC model $name, with its properties, to bdc-$name.bdcm"
    $model = Get-SPBusinessDataCatalogMetadataObject -BdcObjectType "Model" -Name $name -ServiceCc
    Export-SPBusinessDataCatalogModel -Identity $model -Path ".\bdc-$name.bdcm" -ModelsIncluded -P
}
Write-Host "Listing the farm's server-to-server trusts in trusts.txt"
Get-SPTrustedSecurityTokenIssuer | Format-List * | Out-File .\trusts.txt
```

Parsed, not yet run: try it on a test farm first. The rest comes from the mail settings, the source of every customisation that calls out (Chapter 15) and, we would add, the firewall's record of the servers' outbound connections. Each integration then gets a row: both ends, direction, mechanism, schedule, volume, today's identity and an owner for each end.

Keep, re-point, rebuild or retire

Keep an integration when its address, identity and mechanism all hold at the destination. Only Subscription Edition allows it: a Windows account calling a web application that keeps its URL and Windows sign-in (Chapter 21 decides the zones), or a BCS model whose external system accepts the new farm. Online, everything signs in through Entra ID, so nothing is kept.

Re-point it when the mechanism exists at the destination and only the address or the identity changes: an application on CSOM or REST moving to the tenant with an Entra ID registration, a mail relay that accepts the new servers, a script on a new host.

Rebuild it when the mechanism does not exist there. Online: an external list rebuilt in Power Apps, which reaches on-premises data "through the on-premises data gateway" (BCS retirement, above); a remote event receiver rebuilt as a webhook, which cannot cancel the change that triggered it (retirement FAQ, above); Secure Store credentials moved to Azure Key Vault; a server-side script rewritten for the client APIs. On Subscription Edition: anything that signs in with Basic.

Retire it when nothing uses it, its other end has gone, or the owner of that end releases it in writing.

We recommend recording each disposition with the name of whoever owns the other end and the date they agreed: each one changes a system the migration team does not run, on a calendar it does not set.

Choose the identity before the address

Online, app-only access to SharePoint's REST and CSOM endpoints needs a certificate: other options "will result in an Access Denied message" (Entra ID app-only, checked on 6 November 2026). Principals created through `appregnew.aspx` since December 2024 appear in Entra ID as ordinary application principals, which "did not convert them" (retirement FAQ, above), so each integration gets a new registration and its permissions again. Microsoft names Graph the first choice (same FAQ), and its Selected scopes grant access resource by resource: an application consented for them "would initially have no access" (Selected permissions, checked on 6 November 2026). We would give each integration the narrowest grant its job names, as Chapter 17 does for elevated code.

On Subscription Edition, a Windows account keeps its identity in the content under Windows claims (Chapter 21). A server-to-server trust does not travel: its certificate is trusted again with `New-SPTrustedRootAuthority` and registered again with `New-SPTrustedSecurityTokenIssuer` (high-trust add-ins, checked on 6 November 2026), and its app principal is registered in the App Management service (Register-SPAppPrincipal, checked on 6 November 2026), whose database moves where Chapter 19 moves it for workflows. Otherwise we would register the principal again on the rehearsal farm, grant its permissions and test it.

Test each integration before cutover

Prove each integration where it will run, under the identity it will use (the rehearsal farm for an attach, the copy the passes populated online), so that cutover is the day integrations switch, never the day they are first tried (Chapter 9). Four results go beside it:

- **It reads** what it read on the old farm, for the same request.
- **It writes**, and the item shows the Modified By its owner expects.
- **It is refused** under an identity without the grant.
- **It ran its real job** on its real schedule, and the other end's log agrees.

Online, an application over its limits receives HTTP 429 or 503 with a `Retry-After` header, and Microsoft does not allow load testing (throttling, checked on 6 November 2026): test the real job at its real volume, with code that waits as told.

From the moment the source is read-only, every integration that writes to it fails: pause each writer or queue its work, and record what it did not write. After cutover its writes land on the new estate, and a rollback must account for them, as Chapter 9's table sets out.

Out of scope

Sign-in methods and zones are Chapter 21, add-ins installed in sites Chapter 18, what search crawls Chapter 11, hybrid Chapter 25 and the attach Chapter 23. Entra ID, Power Apps and Graph tutorials, integration platforms and licensing are not covered. The site's page on SharePoint Add-ins and Azure ACS sets out what that retirement stopped.

Before you continue

- [ ] Every integration is listed with both ends, direction, mechanism, schedule, volume and today's identity
- [ ] Every front end's logs were read over at least a month, and every account in them has an owner
- [ ] BDC models and trusts are exported, Secure Store's target applications listed, and its passphrase in hand
- [ ] Each integration is kept, re-pointed, rebuilt or retired, with its reason, and the owner of its other end agreed
- [ ] Each one names its destination identity; nothing bound online relies on ACS, an add-in, BCS, Secure Store or a user's password
- [ ] Each one read, wrote, was refused without its grant and ran its real job against the destination before cutover
- [ ] Every integration that writes has a plan for the read-only freeze