PH7X SYSTEMS ®

0.1.0 · CHAPTER 15

Inventory and decide every customisation

SharePoint Migration Field Guide · Chapter 15

pH7x Systems

2026-11-01

15. Inventory and decide every customisation

> What the farm has installed and what the sites use are two lists. The work is in the difference.

A database attach carries the content and leaves the customisations behind. Each one is installed again on the new farm, changed first, replaced or left behind, and that choice is only as good as the inventory under it. An inventory taken only from the farm is one list; the content databases keep a second list of their own, and nothing makes the two agree.

> **Which customisations do your content databases depend on, and what happens to each one on the way to Subscription Edition?**

The decision this chapter leads to is a disposition for every customisation (keep, adapt, replace or eliminate), recorded with its reason, its evidence and its owner.

Which customisations this chapter is for

Everything a SharePoint Server 2016 or 2019 farm has beyond the product: Microsoft's list of what to reinstall before an attach names custom site definitions, style sheets and images, web parts, web services, features and solutions, assemblies, web.config changes, and the InfoPath form templates and data connection files an administrator approved (upgrade content databases, checked on 1 November 2026). The same page notes that Subscription Edition "continues to use 16 as its major version number including in the file system paths", so a package built for 2016 or 2019 targets the same paths; whether its code still works is what the rehearsal farm shows.

Code that runs with elevated privileges gets the audit in Chapter 17 as well as a row here.

Three readings, from two farms

READING	WHAT IT SHOWS	HOW, AND WHERE
Installed	Packages, features at farm and web application scope, files on each server, and each web application's web.config	`Get-SPSolution` ; `Get-SPFeature` at farm and at web application scope; a file comparison of each server against a clean one; the web.config files, their SafeControls and their unique settings. On the 2016 or 2019 farm
Used	The templates, features and web parts each site uses	`stsadm -o enumallwebs` with the includefeatures and includewebparts parameters, for each content database. On the 2016 or 2019 farm, because Subscription Edition removed `stsadm.exe`
Missing on the target	Everything the content references that the new farm does not have	`Test-SPContentDatabase` against a copy of each database, on the rehearsal farm, before anything custom is installed there

Sources for the table: upgrade content databases, Get-SPFeature, Test-SPContentDatabase, what is deprecated or removed and Microsoft's 2013 trial upgrade guidance, which adds the server comparison and the SafeControls, all checked on 1 November 2026. `Get-SPFeature` returns "the enabled Features at each level" of the scope it is given. `Test-SPContentDatabase` checks a database against a web application without changing it, works on a database that is not attached, and can load the database while it runs, which is why it runs on copies.

The first two readings, from an elevated SharePoint Management Shell on the old farm:

```
# Reads only. Writes solutions.csv and one enumallwebs file per content database.
Write-Host "Listing installed solutions, then the webs of each content database"
Get-SPSolution | Export-Csv .\solutions.csv -NoTypeInformation
foreach ($db in Get-SPContentDatabase) {
    stsadm -o enumallwebs -databasename $db.Name -includefeatures -includewebparts |
        Out-File ".\webs-$($db.Name).xml" -Encoding utf8
}
```

What the differences say

WHAT YOU FIND	WHAT IT MEANS	WHAT FOLLOWS
Installed, used by no site	It survived because nobody removed it	A candidate to eliminate, confirmed with its owner before the move
Used and installed	The ordinary case	Decided with the evidence below
Used, but missing on the old farm too	It is broken today, on pages people already open	Fixed or removed at the source; never carried across as an error
Outside every content database: web.config, timer jobs, services, scripts	Only the installed reading sees it	An owner who can say what it does, or it is eliminated

The fourth row is the one the content cannot reveal. A timer job or a web.config setting leaves no trace in the sites that depend on it, so `Test-SPContentDatabase` cannot report it, and only the farm-side reading and a named owner find it.

Decide each one

Microsoft's 2013 guidance on planning for customisations asks the questions that still decide the first cut: "Does it serve a useful business need?", "Is it widely deployed and used?", "Does it do something that you can't do with standard features in the product?" (plan for customisations, checked on 1 November 2026). The answers, and the readings above, give each component its disposition:

- **Keep** when it is used, still needed, and supported as built, its source code and build (or the vendor's version for Subscription Edition) exist, and nothing it depends on was removed.

- **Adapt** when it is needed but something fails: it calls `stsadm.exe`, or relies on PerformancePoint Services or the Claims to Windows Token Service, all three removed from Subscription Edition, which also "replaced Windows Identity Foundation (WIF) 3.5 with WIF 4.5" (what is deprecated or removed, checked on 1 November 2026); or it elevates without the answers Chapter 17 requires.
- **Replace** when the need remains and a standard feature, or another host, now meets it better; where that host should be is our analysis of farm solutions, the SharePoint Framework and external services.
- **Eliminate** when nobody uses it, when only sites that are themselves being retired use it, or when it repeats something the product already does.

Two kinds are never kept as they are. Microsoft's same guidance names "predefined files, features, or site definitions that were changed" and "SharePoint databases that were changed, either by directly changing data or changing the schema" as unsupported, to be replaced before an upgrade. And a component somebody else wrote goes back to its supplier first; Microsoft's trial-upgrade guidance points out that "an upgraded version may be available".

Administration scripts are customisations too. Subscription Edition installs its cmdlets "via a PowerShell module instead of a PowerShell snap-in", and they "will not be compatible with PowerShell Core 6.x or PowerShell 7.x" (what's new in Subscription Edition, checked on 1 November 2026), so every scheduled script runs on the rehearsal farm before anyone relies on it.

We recommend recording each disposition with its reason and its evidence, and having the component's owner agree it in writing: a decision taken by whoever runs the upgrade, alone, is the one that gets reopened on the first Monday. The Subscription Edition migration pack generates the inventory with those columns.

Out of scope

Elevated code is Chapter 17, client-side code Chapter 16, add-ins Chapter 18, workflows Chapter 19, forms Chapter 20, authentication providers Chapter 21 and integrations Chapter 22. Installing what you keep and attaching the content is Chapter 23, where `Test-SPContentDatabase` runs again and has to come back empty, or name only what was decided.

Before you continue

- [ ] Solutions, farm and web application features, server files and every web.config are listed from the old farm
- [ ] Each content database's templates, features and web parts are listed with the sites that use them
- [ ] `Test-SPContentDatabase` ran on copies, on a rehearsal farm with nothing custom installed, and every item it named is in the inventory
- [ ] Installed but unused, and used but already missing, are two separate lists
- [ ] Everything outside the content databases has an owner who can say what it does
- [ ] Each component names its source code and build, and what it depends on that Subscription Edition removed
- [ ] Each component is kept, adapted, replaced or eliminated, with its reason, and its owner agreed