Refer to the Setup documentation for the steps to create a new DGP environment in a location. Once a location has been set up, then the software of the various tiers can be deployed using the CI-CD-CT processes.

DGPDrive Application Deployment Steps

All DGPDrive software is built using the .NET Framework 4.8.1, and uses the XCOPY folder deployment mechanism of the .NET Framework assemblies. Using this mechanism, apps and services are deployed by copying a folder containing the .NET assemblies to the local storage of the target computer. Uninstalling a DGP app is done by deleting the folder. The date of the build is used as the version number, and each new version folder is added as a subfolder under the parent folder of the respective application or service.

IMPORTANT NOTE: the Git repository is configured to ignore all *.config files when synchronizing code in Visual Studio. A generic sample of each .config file is included in the source code as *.config.template. Create a copy of the file without the ".template" extension, and then edit the generic entry values to match a given environment of the location.

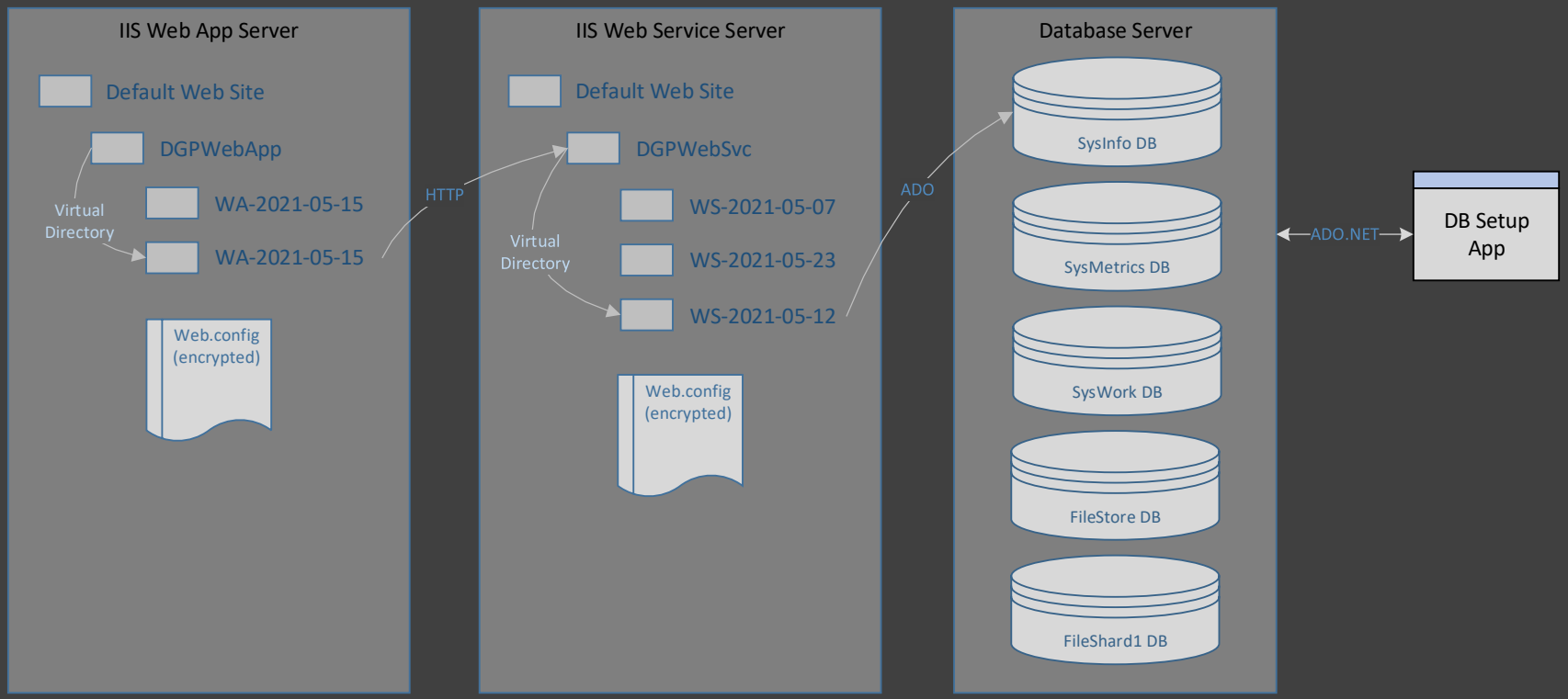
These folders are the parent directories that will be used for deployments of specific versions of their respective DGP apps:

1. On the computer used for IIS Web Services
    a. Create a folder \Inetpub\wwwroot\DGPWebSvc
        i. Copy the assembly folder dgp_websvc_date below the parent folder (add it to the collection of other version subfolders – do not change or delete previous subfolders)
        ii. Edit the Web.config file, or copy it from a previous version
        iii. Edit the path of the web service virtual directory to use the content of the new date subfolder
        iv. Rollback: edit the VM path to use the previous date subfolder and delete the failed date subfolder

2. On the computer used for IIS Web Apps
    a. Create a folder \Inetpub\wwwroot\DGPWebApp
        i. Copy the assembly folder dgp_webapp_date below the parent folder (add it to the collection of other version subfolders – do not change or delete previous subfolders)
        ii. Edit the Web.config file, or copy it from a previous version

iii. Edit the path of the web app virtual directory to use the content of the new date subfolder
iv. Rollback: edit the VM path to use the previous date subfolder and delete the failed date subfolder

3. On the computer used for Client Applications
    a. Create a folder DGP_ClientApps
    b. Create a subfolder DGP_ClientApps\DGP_AutoWork
        i. Copy the assembly folder dgp_autowork_date below the parent folder (add it to the collection of other version subfolders – do not change or delete previous subfolders)
        ii. Edit the App.config file, or copy it from a previous version
        iii. Create a shortcut to the .exe of the new version, or edit a previous shortcut path
        iv. Rollback: edit the shortcut to use the path to the previous .exe and delete the failed date subfolder
    c. Create a subfolder DGP_ClientApps\DGP_DBSetup
        i. Copy the assembly folder dgp_dbsetup_date below the parent folder (add it to the collection of other version subfolders – do not change or delete previous subfolders)
        ii. Edit the App.config file, or copy it from a previous version
        iii. Create a shortcut to the .exe of the new version, or edit a previous shortcut path
        iv. Rollback: edit the shortcut to use the path to the previous .exe and delete the failed date subfolder
    d. Create a subfolder DGP_ClientApps\DGP_Drive
        i. Copy the assembly folder dgp_lattice_date below the parent folder (add it to the collection of other version subfolders – do not change or delete previous subfolders)
        ii. Edit the App.config file, or copy it from a previous version
        iii. Create a shortcut to the .exe of the new version, or edit a previous shortcut path
        iv. Rollback: edit the shortcut to use the path to the previous .exe and delete the failed version subfolder



Each published build of a .NET web app/service produces a folder containing the assembly and any other files it needs to run under the .NET framework installed on each host. These build folders are named for the date they were published. Deployment consists of copying the entire build folder below the directory of the web app or web service on the web server. The web.config file of each app or service contains URL endpoints, ADO.NET connection strings, etc. The web.config of the latest date folder must be modified to include the correct endpoint configurations, usually by copy/pasting them from a previous version. NOTE: In some environments, sections of the web.config file will be encrypted to protect sensitive data using Microsoft's ASPNET_REGIIS.exe utility.

Once the latest date folder web.config file has been edited correctly, the physical path of the web app/service virtual directory is changed to point to that folder. A full regression test is then run to verify all funcitonality. If any problems are encountered, the deployment is rolled back by changing the virtual directory path back to the previous folder, and deleting the failed build folder.