Database Server (SQL Server)

Setting up a new DGP Lattice installation starts with the SQL Server database server, and works its way up through the tiers to the client applications. For development computers, SQL Server Express is all that is needed. In spite of its limitations, the Express edition is also suitable for small scale systems as well, with an easy upgrade path to SQL Server Standard edition when needed.

SQL Server Setup

Refer to Microsoft's documentation for SQL Server installation.

[https://docs.microsoft.com/en-us/sql/database-engine/install-windows/install-sql-server?view=sql-server-ver15]

Most of the default options are used by DGP, with the exception of Mixed Mode database engine configuration:

- New SQL Server stand-alone installation
  - Install Rules
  - Install Type (new instance)
  - Feature Selection
    - Database Engine Services (defaults)
    - Shared Features (defaults)
  - Instance Configuration (Default Instance)
    A named instance can be used, but will affect the ADO.NET connection strings
  - Database Engine Configuration (Mixed Mode)
    Using SQL Server accounts for data access provides better security than using Active Directory domain accounts. Create a password for the "sa" account and store it in a password management system.

The default collation values and other configuration options are used in DGP. Using other options has not been tested, but most of the other options should not result in any problems for DGP systems.

Once SQL Server is installed, use SQL Server Management Studio (SSMS) to create the following 6 empty databases if they do not already exist:

- Hostname_SysInfo
- Hostname_SysMetrics
- Hostname_SysWork
- Hostname_FileStore
- Hostname_FileShard1
- Hostname_FileShard2

Naming conventions will vary greatly from one organization to another. DGP differs from most systems in that:

1. There are no default names, accounts, paths, etc. in a DGP system. Instead, pretty much everything is configurable.
2. Default admin accounts, admin passwords, etc. are supposed to be reset by the administrators of a system, but in many/most cases – they are not. DGP solves this problem once and for all by forcing admins to create their own super admin account name, password, server names, app names, and so on as part of the set up.
3. This also allows for the parts of a DGP system to use any existing naming conventions, rather than impose default DGP names.

There are many options when creating databases, such as where the .mdf and .ldf files will be stored, which Filegroups will be used, what collation will be used, and so on. The choices made for these options depend a great deal on which environment the databases belong to. A developer will just use all the default values to quickly set up the databases on their development workstation, and use the local SQL Server "sa" account in their connection strings (all of which is fine), but none of those shortcuts are acceptable when setting up a QA or production environment.

Therefore, the only rule is that the name of each database must be unique in all of the locations for that environment. To avoid confusion, it is a very good idea to create a naming convention for unique names that also differentiate between the different systems, environments and locations. By default, the names of host servers must already be fairly unique in most cases, so a simple schemaname_hostname convention works pretty well. Short abbreviations for systems, environments and locations can be added as necessary. The objective in the naming convention is to try to keep the names as short as possible while still being human-readable.

DB Setup Utility

Once the empty DGP databases have been created, the DB Setup utility is then used to both build and maintain the tables in each database, along with the core security data in the SysInfo database tables. These processes are idempotent, and will only perform an action the first time they are run. After that, actions that have already been done are skipped, guaranteeing "once and only once" execution of all the schema and data maintenance steps.

When the DB Setup utility is run for the first time, the admin user will be creating important configuration data – especially for the SysInfo database. This includes the DGP account name and password for the system admin account, as well as the service encryption key and version label. This information must be used for all locations of a given environment in a system, since merge replication will synchronize the data between those locations (each environment has its own set of DGP databases, so admin accounts and encryption keys are different between environments of the same system). This data must be saved securely using some sort of system such as a password manager.

The DB Setup utility also creates and displays the ADO.NET connection strings for each database in the location. This information also needs to be saved securely, because those connection strings, the encryption key and key version will all be pasted into the Web.config file of the DGP web services. The encryption key and key version are shared identically among all locations of an environment, while the connection strings are only valid within a single location.

Refer to the Client App setup documentation under Setup for instructions to install the DB Setup utility.

Refer to the DB Setup documentation under Client Tier for instructions on how to use the DB Setup utility.