**PopMedNet CNDS Build:**

Building PMN and CNDS for Development and Deployment

## 

Version 2.0

January 25, 2018

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Revision | Description | Modified by |
| October 30, 2017 | 1.0 | First draft | Jeffrey Ladd |
| January 25, 2018 | 2.0 | Edits to provide more context | Jessica Malenfant |

Contents

[Background 4](#_Toc504652699)

[PopMedNet-CNDS Architecture 4](#_Toc504652700)

[Minimum Tool and Framework Requirements 6](#_Toc504652701)

[Database Server 7](#_Toc504652702)

[Building the Web Portal 8](#_Toc504652703)

[Building the DataMart Client and Adapters 8](#_Toc504652704)

[Building the DataMart Client Installer 8](#_Toc504652705)

[Deploying PopMedNet to IIS 9](#_Toc504652706)

[Physical Metadata Model 10](#_Toc504652707)

# Background

[PopMedNet](https://www.popmednet.org/)™ (PMN) software application currently enables creation, operation, and governance of distributed health data networks. It supports distributed within-network querying for several large-scale health data networks.

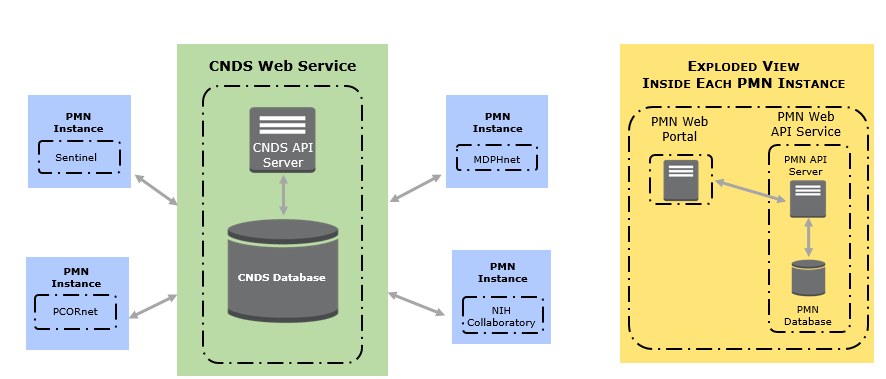
The Cross Network Directory Service (CNDS) extends PMN’s existing functionality to enable cross-network discovery of potential collaborators and data sources and querying of those sources while enforcing governance rules. A flexible data model to capture the metadata collected was developed and integrates CNDS with PopMedNet. As such, a PopMedNet instance needs to be deployed to use CNDS.

# PopMedNet-CNDS Architecture

To minimize the impact on existing networks, CNDS is built gently on top of PMN. CNDS leverages the existing PMN application. This is achieved by implementing CNDS as a set of services that can be invoked by PMN instances, rather than by wholesale modification of PMN. CNDS provides a standard set of functions that PMN can call upon through application programming interfaces (APIs).

Figure 1 shows the high-level architecture of CNDS in relationship to PMN. Each PMN network has its own instance of PMN with the same functionality. The orange box shows an exploded view of what is in each PMN instance (an instance is a stand-alone installation of the application on a server). PMN networks can easily be made part of CNDS by loading the PMN user, organization, and data source information into the CNDS database and configuring the network’s instance to have access to the CNDS API.

Figure 1: PMN - CNDS Integrated Design



Through its APIs, CNDS offers functionality to:

* Process registration requests
* Capture metadata describing users, organizations, registries/research data sets, and queryable data sources
* Enable users to search the metadata in order to explore characteristics of electronic healthcare data sources and identify potential collaborators across and outside of existing PMN distributed data networks
* Route requests and responses across networks

The PopMedNet application is composed of 3 web components:

1. Portal – This covers the UI and supports the legacy PMN components.
2. API – This is the web service that supports all PMN functionalities. (Also known as DNS API.)
3. CNDS API – This is the web service that supports all CNDS functionalities.
4. Resources – This contains the shared files and scripts. In theory, this can be shared by multiple PMN instances with the same version, but it has not been put in practice.

# Minimum Tool and Framework Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Tools and Frameworks** | **Ideal** | **May Work** | **Note** |
| Visual Studio | 2017 | 2015 |  |
| Web Server (IIS) | 8.5 or newer  or use Visual Studio built-in IIS Express | 7.5 | 1. You will need a wildcard SSL certificate for your domain. (\*.YourDomain.com)    * This is because there will be multiple sites that intercommunicate. 2. You will need to add the following Roles and Features to the webserver: |
| .NET Framework | 4.6.1 | 4.6.1 |  |
| Entity Framework | 6 | 6 |  |
| Razor Generator |  |  | Install I<https://github.com/RazorGenerator/RazorGenerator> |
| TypeScript | 2.1.5 | 2.1.5 | https://www.typescriptlang.org |
| ASP.net MVC 4 |  |  | https://www.microsoft.com/en-us/download/details.aspx?id=30683 |
| WIX Toolset | 3.11 | 3.11 | http://wixtoolset.org/releases/ |

## Database Server

|  |  |  |  |
| --- | --- | --- | --- |
| **Tools and Frameworks** | **Ideal** | **May Work** | **Note** |
| SQL Server | 2012 | 2008 R2 | 1. You will need to enable the Filestream and Common Language Runtime features.    * How to enable Filestream: <http://msdn.microsoft.com/en-us/library/cc645923.aspx>    * How to enable CLR: <http://msdn.microsoft.com/en-us/library/ms131048.aspx> 2. We recommend using [Transparent Database Encryption](https://msdn.microsoft.com/en-us/library/bb934049.aspx) ***if*** you have “encryption at rest” requirements. |

# Building the Web Portal

To build a complete PopMedNet-CNDS deployment package, you need to build the PMN Web Portal and the DataMart Client.

The Web Portal consists of 4 components. Using Visual Studio, set the Build Configuration to Debug or Release:

1. Resources: Open and build Lpp.Pmn.Resources solution
2. CNDS: Open and build Lpp.CNDS.sln
3. API: Open and build Lpp.Dns.Api solution
4. Portal: Open and build DistributedNetworkSolution solution

Visual Studio deposits the binary executable in the “bin” folder of the respective solution.

At this point, you can run PopMedNet-CNDS locally within Visual Studio assuming that you have proper PMN and CNDS databases and that the connection strings are setup properly. Use the ConnectionStrings.config-template in Lpp.CNDS, Lpp.Dns.Api, and Lpp.Dns.Portal (DistributedNetworkSolution) to create ConnectionStrings.config files.

# Building the DataMart Client and Adapters

The DataMart Client consists of the DataMart Client Core and the Adapter Packages. Using Visual Studio, set the Configuration to Debug or Release:

1. Open Lpp.Adapters solution
2. Build Lpp.Adapters
3. Execute the /Build/PackageAdapters.ps1 script which will zip all the adapters and place them in the /Lpp.Dns.Api/App\_Data Folder

# Building the DataMart Client Installer

1. Open Lpp.Installers solution
2. Build Lpp.Installers

# Deploying PopMedNet to IIS

The 4 components of the PMN Web Portal are built to their respective “bin” folder. The Adapter packages are built, packaged, and deployed directly to Lpp.Dns.Api/App\_Data on successful build of each adapter project. The DataMartClient installer folder should have been copied to Portal/DataMartClient.

To deploy PopMedNet, publish the following:

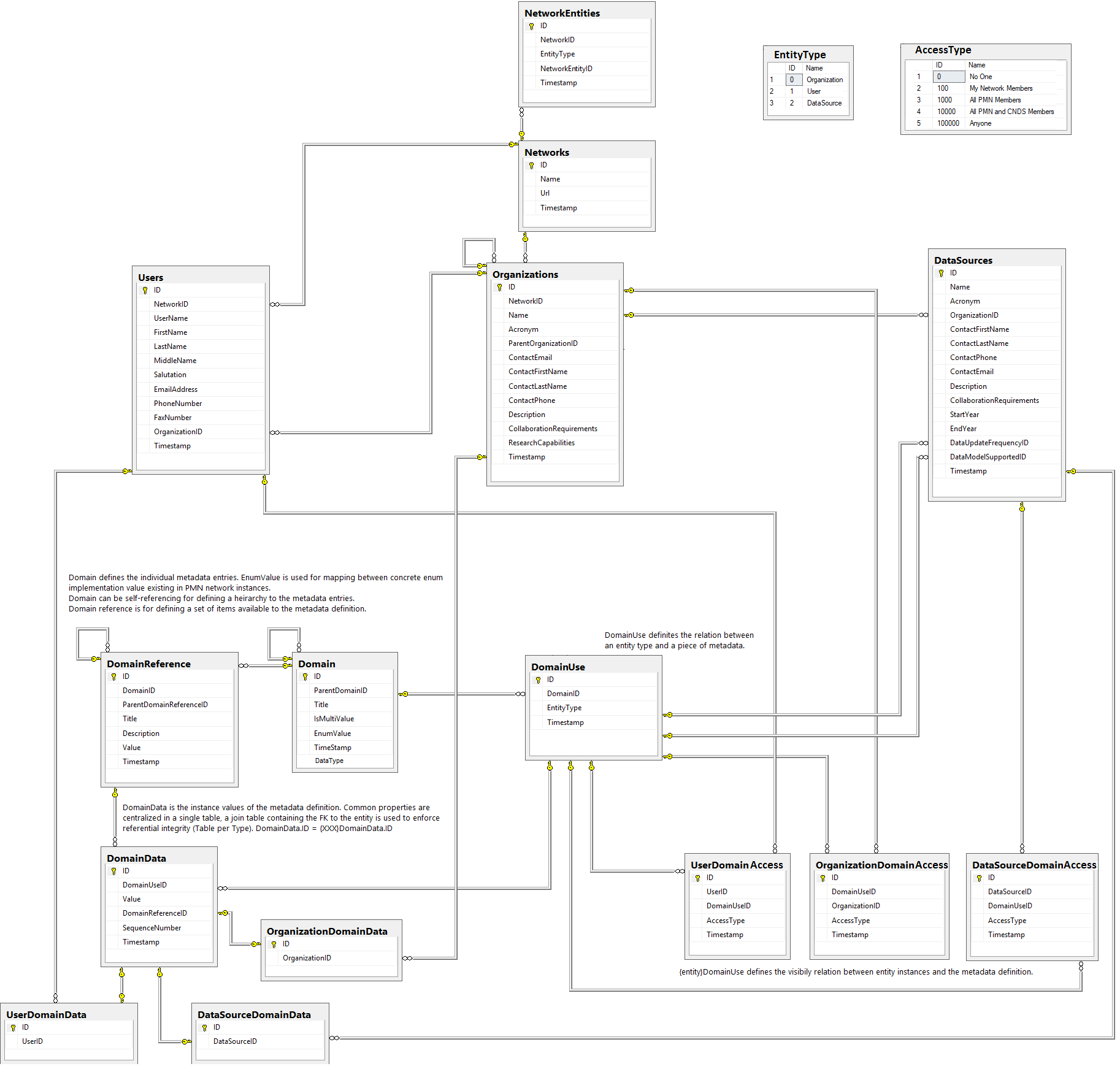
1. Resources: Right click on Lpp.Pmn.Resources and select Publish
2. CNDS: Right click on Lpp.CNDS.API and select Publish
3. API: Right click on Lpp.Dns.Api and select Publish
4. Portal: Right click on Lpp.Dns.Portal and select Publish
5. DataMartClient Installer: Open the Lpp.Installers and compile, then go to the bin folder and copy the MSI to the published location of the Portal and place it in a folder called DataMartClient

Select Custom and create Profile in each case. You can choose to deploy directly to IIS or to a local folder.

After deploying Edit the Web.Configs for the following values

1. Portal
   1. CurrentTheme
   2. ContactUsEmail
   3. NotificationSubjectLine
   4. NetworkMessageSubjectLine
   5. NetworkMessageSignoffFrom
   6. ServiceUrl
   7. ResourceUrl
   8. Log4Net Configurations
2. API
   1. ResetPasswordUrl
   2. CNDS.URL
   3. Log4Net Configurations

# Physical Metadata Model



In the physical metadata model above, the user-friendly table and field names are displayed. The data dictionary below, maps the user-friendly table and field names to the actual names in the database. Table names are highlighted in light blue.

The tables in the data dictionary match the tables in the metadata model reading first top to bottom and then left to right, except the two lookup tables EntityType and AccessType.

|  |  |
| --- | --- |
| **Physical Model Name** | **Database Field or**  **Table Name** |
| **NetworkEntities** | **NTWRK\_ENTY\_T** |
| ID | ID |
| NetworkID | NTWRK\_ID |
| EntityType | ENTY\_TYP\_CD |
| NetworkEntityID | NTWRK\_ENTY\_ID |
| Timestamp | CNDS\_UPDT\_TS |
| **Networks** | **NTWRK\_T** |
| ID | ID |
| Name | NTWRK\_NM |
| URL | URL\_TXT |
| Timestamp | CNDS\_UPDT\_TS |
| **Users** | **USR\_T** |
| ID | ID |
| NetworkID | NET\_ID |
| UserName | USR\_NM |
| FirstName | FRST\_NM |
| LastName | LAST\_NM |
| MiddleName | MID\_NM |
| Salutation | PREFX\_NM |
| EmailAddress | EMAIL\_NM |
| PhoneNumber | PHONE\_NBR |
| FaxNumber | FAX\_NBR |
| OrganizationID | ORG\_ID |
| Timestamp | CNDS\_UPDT\_TS |
| **Organizations** | **ORG\_T** |
| ID | ID |
| NetworkID | NTWRK\_ID |
| Name | ORG\_NM |
| Acronym | ORG\_ACRYN\_TXT |
| ParentOrganizationID | PARNT\_ORG\_ID |
| ContactEmail | CONTC\_EMAIL\_NM |
| ContactFirstName | CONTC\_FRST\_NM |
| ContactLastName | CONTC\_LAST\_NM |
| ContactPhone | CONTC\_PHONE\_NBR |
| Description | ORG\_DSC |
| CollaborationRequirements | COLLB\_REQMT\_TXT |
| ResearchCapabilities | RSRCH\_CAP\_TXT |
| Timestamp | CNDS\_UPDT\_TS |
| **DataSources** | **DATA\_SRC\_T** |
| ID | ID |
| Name | DATA\_SRC\_NM |
| Acronym | DATA\_SRC\_ACRYN\_TXT |
| OrganizationID | ORG\_ID |
| ContactFirstName | CONTC\_FRST\_NM |
| ContactLastName | CONTC\_LAST\_NM |
| ContactPhone | CONTC\_PHONE\_NBR |
| ContactEmail | CONTC\_EMAIL\_TXT |
| Description | DATA\_SRC\_DSC |
| CollaborationRequirements | COLLB\_REQMT\_TXT |
| StartYear | BEG\_YEAR\_NBR |
| EndYear | END\_YEAR\_NBR |
| DataUpdateFrequencyID | DATA\_UPDT\_FREQ\_ID |
| DataModelSupportedID | DATA\_MODEL\_SUPP\_ID |
| Timestamp | CNDS\_UPDT\_TS |
| **DomainReference** | **DOMN\_REF** |
| ID | ID |
| DomainID | DOMN\_ID |
| ParentDomainReferenceID | PARNT\_DOMN\_REF\_ID |
| Title | DOMN\_NM |
| Description | DOMN\_REF\_DSC |
| Value | DOMN\_REF\_CD |
| Timestamp | CNDS\_UPDT\_TS |
| **Domain** | **DOMN** |
| ID | ID |
| ParentDomainID | PARNT\_DOMN\_ID |
| Title | DOMN\_NM |
| IsMultiValue | MULTI\_VAL\_IND |
| EnumValue | PMN\_DOMN\_NM |
| TimeStamp | CNDS\_UPDT\_TS |
| DataType | DATA\_TYP\_CD |
| **DomainUse** | **DOMN\_USE** |
| ID | ID |
| DomainID | DOMN\_ID |
| EntityType | ENTY\_TYP\_ID |
| Timestamp | CNDS\_UPDT\_TS |
| **DomainData** | **DOMN\_DATA** |
| ID | ID |
| DomainUseID | DOMN\_USE\_ID |
| Value | DOMN\_DATA\_TXT |
| DomainReferenceID | DOMN\_REF\_ID |
| SequenceNumber | SEQ\_NBR |
| Timestamp | CNDS\_UPDT\_TS |
| **OrganizationDomainData** | **ORG\_DOMN\_XREF** |
| ID | ID |
| OrganizationID | ORG\_ID |
| **UserDomainAccess** | **USR\_DOMN\_ACCESS** |
| ID | ID |
| UserID | USR\_ID |
| DomainUseID | DOMN\_USE\_ID |
| AccessType | ACCES\_TYP\_ID |
| Timestamp | CNDS\_UPDT\_TS |
| **OrganizationDomainAccess** | **ORG\_DOMN\_ACCESS** |
| ID | ID |
| DomainUseID | DOMN\_USE\_ID |
| OrganizationID | ORG\_ID |
| AccessType | ACCES\_TYP\_ID |
| Timestamp | CNDS\_UPDT\_TS |
| **DataSourceDomainAccess** | **DATA\_SRC\_DOMN\_ACCESS** |
| ID | ID |
| DataSourceID | DATA\_SRC\_ID |
| DomainUseID | DOMN\_USE\_ID |
| AccessType | ACCES\_TYP\_ID |
| Timestamp | CNDS\_UPDT\_TS |
| **UserDomainData** | **USER\_DOMN\_XREF** |
| ID | ID |
| UserID | USR\_ID |
| **DataSourceDomainData** | **DATA\_SRC\_DOMN\_XREF** |
| ID | ID |
| DataSourceID | DATA\_SRC\_ID |
| **EntityType** | **ENTY\_TYP** |
| ID | ID |
| Name | ENTY\_TYP\_NM |
| **AccessType** | **ACCES\_TYP** |
| ID | ID |
| Name | ACCES\_TYP\_NM |