

Created by Acmeware, Inc., all rights reserved.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

COVID-19 Dashboard

Installation and Configuration

Overview

COVID-19 dashboard description

The Acmeware COVID-19 Dashboard is a MEDITECH Data Repository-based report that provides a variety of useful measures for patients with confirmed or suspected COVID-19. It includes relevant metrics like patient census, bed occupancy, ventilator availability, and more. The dashboard is designed to provide you with information to help manage patients and resources during the pandemic.

While the report should be set up and initially configured by an IT / IS system or database administrator, you will need information from other areas in your hospital, based on how your MEDITECH system is configured. Some critical information you'll need to configure the report includes:

1. How many ventilators does your hospital have?
2. What query (or queries) are used to document ventilator usage?
3. What lab tests are used to identify coronavirus and how is a positive result indicated?

As you work through the [Configuration Details](#) section below, keep in mind that some of the information you'll need might come from clinical informaticists in other departments, your lab, etc.

Important!

Successful installation and configuration of the dashboard requires an intermediate level of technical skill with Microsoft SQL Server, Data Repository and Microsoft SQL Reporting Services. Users should be comfortable installing and modifying T-SQL stored procedures and directly editing table data in the user database, as well as modifying and deploying a Visual Studio/SQL Server Data Tools .rdl file to an intranet-based Reporting Services instance. For more information, see the [Technical Prerequisites](#) and [Support](#) sections of this document.

Technical Prerequisites

Items required for installation & use

- 1) The COVID-19 Dashboard is built with Microsoft SQL Server components and can only be installed to work with the MEDITECH Data Repository. It is NOT an NPR, Report Designer, or BCA report.
- 2) A user database on your Data Repository SQL Server to contain the necessary SQL Server stored procedures and functions. If you already have a database for your own custom DR reports, you can install our components in it. If you don't, we recommend creating a new database for this purpose – we do NOT recommend installing our components in your live or test DR databases.
- 3) Security permissions to install and publish the report include dbo or sysadmin access to your SQL user database, and system administrator permissions for SQL Server Reporting Services/Report Manager.
- 4) Approximately 500MB of free disk space on your DR SQL Server.

- 5) Your DR SQL Server version should be SQL 2012 or newer.
- 6) The preferred report delivery method is via Intranet deployment with **SQL Server 2016 Reporting Services** (or newer) with Visual Studio/SQL Server Data Tools 2015 (or newer). If you don't have SQL Report Services, standalone, desktop deployment can be done with Microsoft Report Builder version 15.0 (currently available from Microsoft as a free download).

Report Components

Objects in the report package

The zip file report package includes these objects:

- 1 T-SQL script file, *Covid19TrackingDB.publish.sql*
- 1 report definition file, *COVID-19 Dashboard.rdl*

Once the script file is successfully executed for the first time, it creates 12 tables, 10 stored procedures and 3 user-defined functions in your reporting database. See the [Configuration Details](#) below for more details.

How the Dashboard Works

After setup is complete (following steps in the [Configuration Details](#) section), when the report is executed, the report logic first builds a patient population, including patients discharged within the last 60 days (you can change this setting) as well as current inpatients, including ER and observation patients. Then for these patients, all the relevant clinical items are written to a table which is then subsequently queried by the report and presented to the user.

The report datamart is populated on the first report execution, and presentation may take a minute or more. Subsequent executions (e.g. by different users) present cached data and therefore present faster. If the report hasn't been run by any user within 60 minutes (this refresh value is configurable), the next execution will update the datamart and reset the refresh counter.

Configuration Details

Instructions for installing and configuring the report

1 - Database Setup

1. **Open the *Covid19TrackingDB.publish.sql* file from SQL Server Management Studio.** This one file, after being configured as described, adds all the necessary stored procedures, functions, and tables to your reporting database.
2. **Configure the script variables.** Follow the comments in the script file to set the necessary variables for your system. First enable SQLCMD mode by selecting it from the "Query" menu in SSMS. Once enabled, you should see several lines that start with ":setvar" highlighted. Set the variables for:
 - a. The database name to install the database objects to. This should be a database on the DR server for user-developed reports, NOT the MEDITECH live or test databases. Contact the DBA or SQL Server system administrator if you need a new database created.
 - b. Your MEDITECH platform: MAGIC, CS, or 6X. (6X includes all versions of 6.0, 6.1 and Expanse.)
 - c. The name of your live DR NPR database (e.g., livedb, livendb, etc.)
 - d. For 6.0x and 6.1x/Expanse systems only: the name of your live DR M-AT database (e.g. livefdb). MAGIC and CS platforms can leave this value as "N/A."
3. **Execute the script.** You'll see two result sets in the bottom window; the first is a single-line summary of the default system configuration settings; the second is the output for the report. Until the tables are

configured further (next section) these results can be ignored; but seeing them indicates the procedures are running error-free.

If you receive error messages: if the Messages tab displays errors that reference a MT platform that is different than yours, you can ignore them. For example, non-MAGIC sites can ignore this error:

```

Results Messages
Creating [dbo].[spAW_COVID19_Populate_ConfigDictItems_CS]...
Creating [dbo].[spAW_COVID19_Populate_ConfigDictItems_MAGIC]...
Msg 207, Level 16, State 1, Procedure spAW_COVID19_Populate_ConfigDictItems_MAGIC, Line 311 [Batch Start Line 1694]
Invalid column name 'NomenclatureID'.
Msg 207, Level 16, State 1, Procedure spAW_COVID19_Populate_ConfigDictItems_MAGIC, Line 311 [Batch Start Line 1694]
Invalid column name 'NomenclatureID'.
Creating [dbo].[spAW_COVID19_Populate_DataActivity_6X]...
Creating [dbo].[spAW_COVID19_Populate_DataActivity_CS]...

```

Please see the [Support](#) section below if you receive any other installation errors.

4. Confirm the setup. When the script executes, you should have 12 tables, 9 stored procedures, 1 view and 3 functions in your database. See the [Appendix](#) section for details on these objects.

MyUserRptDb

- Database Diagrams
- Tables
 - FileTables
 - External Tables
 - Graph Tables
 - dbo.tbAW_COVID19_ConfigAppSettings
 - dbo.tbAW_COVID19_ConfigCpts
 - dbo.tbAW_COVID19_ConfigDiagnosis
 - dbo.tbAW_COVID19_ConfigLabs
 - dbo.tbAW_COVID19_ConfigLocations
 - dbo.tbAW_COVID19_ConfigOrders
 - dbo.tbAW_COVID19_ConfigProblems
 - dbo.tbAW_COVID19_ConfigQueries
 - dbo.tbAW_COVID19_DataActivity
 - dbo.tbAW_COVID19_DataPtBasePopulation
 - dbo.tbAW_COVID19_DimDate
 - dbo.tbAW_COVID19_DimNomenclatureCodes
- Functions
 - Table-valued Functions
 - dbo.tvAW_COVID19_MVXml
 - dbo.tvAW_COVID19_XmlToText
 - Scalar-valued Functions
 - dbo.fnAW_COVID19_TextToXml
- Views
 - dbo.vwAW_COVID19_PatientDetails
- Stored Procedures
 - dbo.spAW_COVID19_ConfigAppSettings
 - dbo.spAW_COVID19_GetAppSettings
 - dbo.spAW_COVID19_GetCovidDailyData
 - dbo.spAW_COVID19_GetLocationCensus
 - dbo.spAW_COVID19_Populate_ConfigDictItems_CS
 - dbo.spAW_COVID19_Populate_DataActivity_CS
 - dbo.spAW_COVID19_Populate_DataPtBasePopulation_CS
 - dbo.spAW_COVID19_Populate_DimDate
 - dbo.spAW_COVID19_Populate_DimNomenclatureCodes

2 - Database Configuration

Overview

The setup script builds tables, stored procedures and functions in the user-specified database, including eight “_Config” tables. Ensuring these configuration tables have correct data for your hospital is required for the dashboard’s accuracy. Please note the following standards, which apply to all the “_Config” tables:

1. All but one of the tables are pre-populated with standard data, based on WHO and CDC guidelines, your MIS dictionary values and your existing nomenclature mapping. You should review the records in these tables for accuracy, as well as add any new records as appropriate for your system. Since there is no standard OE/OM workflow for Covid-19, the *tbAW_COVID19_ConfigOrders* table is a placeholder for you to customize as appropriate for your facility.
2. Every table contains one or more columns to indicate whether an item may be an indicator of COVID-19. These will be set to either 1 (true) or 0 (false). By default, only WHO/CDC provided codes (CPTs, diagnoses) will be set to true; most of them will be set to false. These true/false values should be reviewed and updated as appropriate for your facility.
3. Please do not delete any table records that may be inactive or where column values are set to 0 – the system may still refer to these, or they can be activated later if necessary.

Confirm and modify values (if necessary) in the config tables. The steps below describe how to review what’s in each table and where revisions or additions might be necessary. For a detailed description of each table, see the [Appendix](#) section at the end of the document. We recommend making table edits directly using SQL Server Management Studio rather than T-SQL UPDATE statements.

1. **_ConfigAppSettings** – contains the variables set with the setup script and other report-level controls.
Column values to modify: *VentilatorCnt* (enter the number of ventilators available at your facility). You may also modify the default values for *CensusDays*, *DataRefreshMinutes* or the warning and critical rates for bed occupancy and ventilator use.
2. **_ConfigCpts** – add any CPT code values here that may be missing, You can also override the default name with a custom *CptDisplayName* if you wish, and set the *IsCovid19* and *IsIndicatorOfCovid19* values as appropriate (1 = true and 0 = false).
3. **_ConfigDiagnoses** – diagnosis codes should all be reviewed and have the *IsCovid19* and *IsIndicatorOfCovid19* values set as appropriate (1 = true and 0 = false). The default display name can be overridden with any value you enter in the *DxCodeDisplayName* column.
4. **_ConfigLabs** – lab tests should be reviewed and have the *IsCovid19Test* value set as appropriate (1 = true and 0 = false). In addition, the literal values in the *PositiveIndicatorResult* and *NegativeIndicatorResult* columns should be modified if necessary, entering a list of all possible values separated by commas with no spaces. The default test display name can be overridden with any value you enter in the *TestDisplayName* column.
5. **_ConfigLocations** – every location from the MIS location dictionary will appear here; you can set the *ExcludeFromDashboard* switch to true or false (1 or 0) to control which locations appear on the dashboard. (By default, every active location will appear.) If your ICU and ED/ER locations have existing nomenclature mappings, they will be indicated in the *IsIcu* and *IsEr* columns with 1 = true and 0 = false; any of these can be manually overridden as necessary. You can also set the *IsCovid19Only* column to

true or false if you have a location reserved for these patients. Finally there is a column to override the default display name if you wish.

6. **_ConfigOrders** – this optional table can be used if you have Covid-19 specific OE/OM orders; values should be entered manually directly in the table.
7. **_ConfigProblems** – values in this table are based on standard IMO SNOMED mappings; you can set the *IsCovid19* and *IsIndicatorOfCovid19* values as appropriate (1 = true and 0 = false) and override the display name as in the other tables.
8. **_ConfigQueries** – this table is pre-populated from MIS query dictionary entries that contain words like “ventilator,” “respirator” and “Covid.” Since this list may be incomplete, review every value and set the *IsCovid19Qry*, *IsIndicatorOfCovid19Qry* and *IsVentilatorQry* column values to 1 or 0 (true or false) as appropriate. By default, they are all set to false. For the *PositiveIndicatorResponses* and *NegativeIndicatorResponses* columns, enter the text for how a positive (or negative) response is indicated. If there is more than one possible response, enter each value in a comma-separated list with no spaces. For example, if you have positive response values of either YES or CONFIRMED, enter them like this in the *PositiveIndicatorResponses* column: YES,CONFIRMED . **Only account-level queries from MEDITECH applications (like ADM or REG) are included in the dashboard.**

3 - Report Configuration

The dashboard can be configured as an intranet-based report if you use SQL Server Reporting Services, or as a standalone/desktop report (single-user) if you use Microsoft Report Builder. Detailed steps for each are below.

If you have an existing SQL Server Reporting Services instance:

You can either add the “COVID-19 Dashboard.rdl” file to a new or existing Visual Studio/SQL Server Data Tools project, or you can upload it directly to the Report Manager instance on your intranet.

1. To add to an existing project:

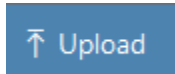
- a. Copy the “COVID-19 Dashboard.rdl” file from where you unzipped the package into the folder and location where your existing .rdl files reside.
- b. From the Solution Explorer pane in VS/SSDT, right-click the “Reports” folder and select “Add...Existing Item.” Navigate to your project folder and select the .rdl file.
- c. Open the COVID-19 Dashboard.rdl file in Design mode. In the Report Data pane, modify the “CovidTrackingDB” data source by double-clicking it and updating the “Connection string” with your DR server name and user database name.

Data Source=**DR1**;Initial Catalog=**MyUserRptDB**

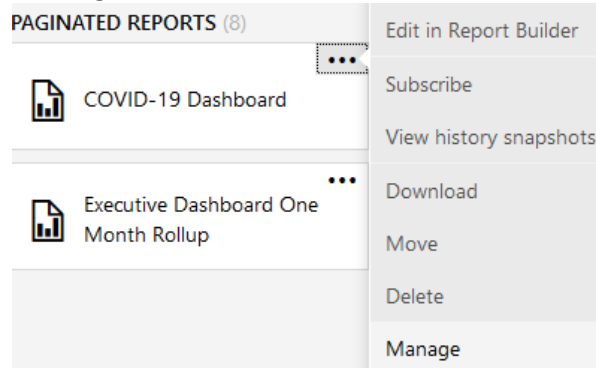
- d. Set the data source credentials by selecting the “Credentials” item from the still-open Data Source properties window. If every user that runs the report already has a SQL login with their Active Directory account (or belongs to a valid AD group) on the DR server, choose “Use Windows Authentication.” If every report user does not have an AD login, select the “Use this user name and password” option and provide the account credentials of a standard SQL login that has permissions to execute the report stored procedures from your user database. Now select “OK.”
- e. Build and deploy the report from VS/SSDT to the Report Server. Note that it will be deployed to the existing folder for your existing VS project.

2. To add the report directly to Report Manager:

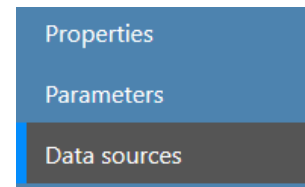
- a. In your browser, navigate to your Report Manager URL and the folder in which you wish to deploy the dashboard.
- b. From the horizontal toolbar, select “Upload” and select the “COVID-19 Dashboard.rdl” file.



- c. After the file is added, click the ellipses (...) to the upper-right of the report name and select “Manage”



- d. Select the “Data sources” option from the left-hand menu.
- e. Modify the data source as detailed in step 1d above, selecting either the “As the user viewing the report” option (which uses Windows AD authentication) or with a dedicated standard SQL Server login and password (with the “Using the following credentials” option).
- f. Click “Save” toward the bottom of the screen.



To use Microsoft Report Builder (single-user desktop installation):

1. Microsoft Report Builder is available for free download and use from [Microsoft’s Downloads page](#).
2. After launching the app, from the menu, select “File...Open” and select the “COVID-19 Dashboard.rdl” file from where you unzipped the package. If you receive a prompt to select a report server, click “Cancel.”
3. In the Report Data pane on the left, expand the “Data Sources” folder and modify the “CovidTrackingDB” data source by double-clicking it and updating the “Connection string” with your DR server name and user database name.

Data Source=**DR1**;Initial Catalog=**MyUserRptDB**

4. Set the data source credentials by selecting the “Credentials” item from the still-open Data Source properties window. There are two options, both of which require a valid SQL Server login, either with an existing Windows AD account (integrated security) or a standard SQL login (not tied to Windows). If your AD account has an SQL login, select “Use current Windows user.” If not, select the “Use this user name and password” option and provide the account credentials of a standard SQL login that has permissions to execute the report stored procedures from your user database. Click “OK.”
3. Click “Run” from the main toolbar to see the report. It may take 1 minute or longer for the report to first execute. The first report run populates the underlying data tables after querying the DR database; this process will take some time, depending on the number of discharge days specified during setup and your overall system performance.

Running the Report (SQL Server Reporting Services only)

From the Report Manager, select the “COVID-19 Dashboard” report. Be patient when the “Loading” indicator appears; it may take 1 minute or longer for the report to first execute. The first report run populates the underlying data tables after querying the DR database; this process will take some time, depending on the number of discharge days specified during setup and your overall system performance.

Subsequent report executions will retrieve data from cache and refresh within seconds. This includes multiple users running the report simultaneously. After the default interval of 60 minutes has passed, the next report execution will repopulate the tables, again taking a bit longer to retrieve data from the DR tables. This refresh interval can be customized as detailed in Step 1 of the [Database Configuration](#) section above.

Support

How to get help

Beyond the scope of this document, Acmeaware will provide limited support to assist with the installation of the dashboard. **Please contact us for support only after you’ve read the entire document**, including the *Technical Prerequisites* and *Configuration Details* sections above.

To contact us for support, please see the [COVID-19 support page](#) on our website.

Existing Acmeaware Partnership or OneView customers should contact us via our [Customer Support Portal](#).

Appendix

A list of dashboard package objects and their description

Items below on the first 2 pages (with white background) are those that require review and potential updates by the IT/IS staff implementing this report. Subsequent items (those with gray background) require no modification after the initial installation.

Database object	Description	Maintenance
tbAW_COVID19_ConfigAppSettings	Configuration settings for the dashboard, including census days, refresh minutes, last refresh status, ventilator count, and warning and critical rates for bed occupancy and ventilator usage.	Populated during initial setup but should be manually updated as needed.
tbAW_COVID19_ConfigCpts	CPT codes and names, along with true/false (bit) switches for <i>IsCovid19</i> and <i>IsIndicatorOfCovid19</i> that trigger display and data items on the dashboard.	Populated during initial setup based on WHO/CDC CPT values and your nomenclature mapping. Update manually as needed or as WHO/CDC guidance changes. <i>CptDisplayName</i> can be entered manually with your preferred description to appear on report.
tbAW_COVID19_ConfigDiagnosis	ICD-10 diagnosis codes and names, along with true/false (bit) switches for <i>IsCovid19</i> and <i>IsIndicatorOfCovid19</i> that trigger display and data items on the dashboard.	Populated during initial setup based on WHO/CDC ICD-10 guidelines and your nomenclature mapping. Update manually as needed or as WHO/CDC guidance changes. <i>DxCodeDisplayName</i> can be entered manually with your preferred description to appear on report.
tbAW_COVID19_ConfigLabs	Lab test print numbers, mnemonics and names, along with a true/false (bit) switch for <i>IsCovid19Test</i> that triggers display and data items on the dashboard. All possible case-sensitive literal values for both positive and negative test results should be entered as a comma-separated list with no spaces and no quote marks.	Populated during initial setup based on WHO/CDC LOINC guidelines and your nomenclature mapping. Update manually as needed or as WHO/CDC guidance changes. <i>TestDisplayName</i> can be entered manually with your preferred description to appear on report.

tbAW_COVID19_ConfigLocations	MIS location dictionary mnemonics, names, bed counts, and location types, along with true/false (bit) switches for <i>ExcludeFromDashboard</i> , <i>IsIcu</i> , <i>IsEr</i> and <i>IsCovid19Only</i> .	Populated during initial setup from MIS location dictionary, including SNOMED-CT mappings for ER and ICU locations. Update manually as needed. By default, inactive locations are set to not appear on report. <i>LocationDisplayName</i> can be entered manually with your preferred description to appear on report.
tbAW_COVID19_ConfigOrders	OE/OM procedures, categories, and names, along with true/false (bit) switches for <i>IsCovid19</i> and <i>IsIndicatorOfCovid19</i> that trigger display and data items on the dashboard.	Requires all manual entries to enable items to appear on report. <i>OrderProcedureDisplayName</i> can be entered manually with your preferred description to appear on report.
tbAW_COVID19_ConfigProblems	MIS problem dictionary problems and names, along with true/false (bit) switches for <i>IsCovid19</i> and <i>IsIndicatorOfCovid19</i> that trigger display and data items on the dashboard.	Populated during initial setup based on WHO/CDC SNOMED-CT guidelines and your nomenclature mapping. Update manually as needed or as WHO/CDC guidance changes. <i>ProblemDisplayName</i> can be entered manually with your preferred description to appear on report.
tbAW_COVID19_ConfigQueries	MIS query mnemonics and descriptions, along with true/false (bit) switches for <i>IsCovid19Qry</i> , <i>IsIndicatorOfCovid19Qry</i> and <i>IsVentilatorQry</i> that trigger display and data items on the dashboard. All possible case-sensitive literal values for both positive and negative query responses should be entered as a comma-separated list with no spaces and no quote marks.	Populated during initial setup with wildcard matches on keywords: COVID, SARS-2, CORONAVIRUS, VENTIL. Review all entries in this table and update as needed. <i>QueryDisplayName</i> can be entered manually with your preferred description to appear on report.
tbAW_COVID19_DataActivity	Table for storing and presenting compiled patient data. Refreshes based on interval specified in <i>tbAW_COVID19_ConfigAppSettings</i> .	None.
tbAW_COVID19_DataPtBase Population	Table for storing and presenting base patient population. Refreshes based on interval specified in <i>tbAW_COVID19_ConfigAppSettings</i> .	None.
tbAW_COVID19_DimDate	Dimension table for user-friendly presentation of dates.	None.

tbAW_COVID19_DimNomenclature Codes	Dimension table built on initial setup with WHO and CDC standard nomenclature values.	None.
spAW_COVID19_ConfigAppSettings	Creates ConfigAppSettings table and inserts default settings and user-parameter values.	None.
spAW_COVID19_GetAppSettings	Retrieves dashboard application settings at runtime.	None.
spAW_COVID19_GetCovidDailyData	Retrieves current data (or refreshes tables) at runtime for report display.	None.
spAW_COVID19_GetLocationCensus	Retrieves patient census data for display by location.	None.
spAW_COVID19_Populate_Config DictItems	Populates configuration tables with default WHO/CDC nomenclature values during initial setup.	None.
spAW_COVID19_Populate_Data Activity	Builds primary dataset for dashboard display by retrieving Covid-19 indicators for base patient population.	None.
spAW_COVID19_Populate_DataPt BasePopulation	Builds base patient population for use with Covid-19 indicators.	None.
spAW_COVID19_Populate_DimDate	Populates date dimension table during initial setup.	None.
spAW_COVID19_Populate_DimNomenclatureCodes	Populates nomenclature dimension table with WHO/CDC standard values.	None.
tvAW_COVID19_MVXml	Parses multi-value parameters in SSRS.	None.
tvAW_COVID19_XmlToText	Converts XML to text.	None.
fnAW_COVID19_TextToXml	Converts text values to XML.	None.
vwAW_COVID19_PatientDetails	Retrieves base patient population for display on census report.	None.
Report	Description	Maintenance
COVID-19 Dashboard.rdl	Report definition file for the COVID-19 dashboard.	After initial data source modification, none.