



CONFIDENTIAL - Do not distribute.

Introduction to Reporting with Civis Platform

Additional Resources

For more in-depth information please see these materials:

- Tableau Reports
 - [Publishing Reports in Tableau](#)
 - [Troubleshooting Common Problems](#)
- Civis Platform Services
 - [Civis Service Deployment](#)
 - [Advanced Deployments](#)
 - [R Shiny App Deployments](#)
- Google Data Studio
 - [Civis Platform Google Data Studio Connector](#)
- Email Notifications
 - Code examples for posting Job outputs in [Python](#) and [Javascript](#) for use in Markdown tables

Today's Agenda

- [Reporting Best Practices \[Video\]](#)
 - [Report Viewer accounts \[Video\]](#)
 - [The Reporting Process \[Video\]](#)
- [Reports on Civis Platform \[Video\]](#)
 - [Reports are Platform Objects \[Video\]](#)
 - Tableau
 - [Prerequisites and Connecting to Data \[Video\]](#)
 - [Publishing to the Civis Tableau Server \[Video\]](#)
 - [HTML](#)
 - [HTML Reports \[Video\]](#)
 - [R markdown \[Video\]](#)
 - [Services \[Video\]](#)
- [Other Reporting Methods with Civis Platform \[Video\]](#)
 - [Google Sheets \[Video\]](#)
 - [Google Data Studio](#)
 - [Overview \[Video\]](#)
 - [Setup \[Video\]](#)
 - [Job Notification Emails \[Video\]](#)

Reporting Best Practices

Report Viewer Accounts

- Utilizing Report Viewer Accounts is an excellent way to provide team members with the ability to view Tableau and HTML reports in Platform.
- These report viewer accounts simply allow your colleagues to view reports in Civis Platform and do not provide them with access to the underlying data or the ability to create Platform jobs (scripts, import, exports, etc).
- To request a new Report Viewer account, contact your Organization Admins.

The Reporting Process



- | | | | |
|---|--|---|--|
| <ul style="list-style-type: none">- What questions are you trying to answer?- Who is your target audience?- What tables/views house the data you can build your metrics from?- Are there tables you don't have permission on that you need to request access on? | <ul style="list-style-type: none">- Write the needed code to retrieve, clean, load data into your database- If the table building process incorporates several tasks, consider building a Platform Workflow to chain tasks together- Trigger this process when source data is updated or use automation to meet the data refresh cadence | <ul style="list-style-type: none">- Chose a reporting method and build your report to best convey in information in your reporting tables- Consider: development time and difficulty as well as effort to maintain, impact, and ease of use for the report viewer- Using the data sources, develop your Report in your chosen framework | <ul style="list-style-type: none">- Make sure to Share the Report via the standard action menu with the Users and Groups that you want to be able to view this.- How often should your report update? Schedule your refresh script on this cadence.- Consider using this guideline: your report should refresh double the cadence of your slowest updating data source |
|---|--|---|--|

Ultimately, steps 2 and 4 can be configured in a Platform Workflow to automate the updating of your report

Reports on Civis Platform



Reports On Civis Platform

Three types of reports

Tableau

- Backed by a Tableau Workbook Published to the Civis Tableau Server

HTML

- Backed by a HTML document defining the page with Platform Javascript support

Civis Services

- Backed by code in a Git repository and deployed as a web app using your Civis compute resources

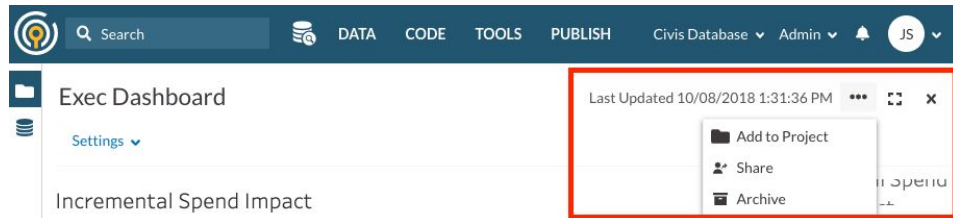
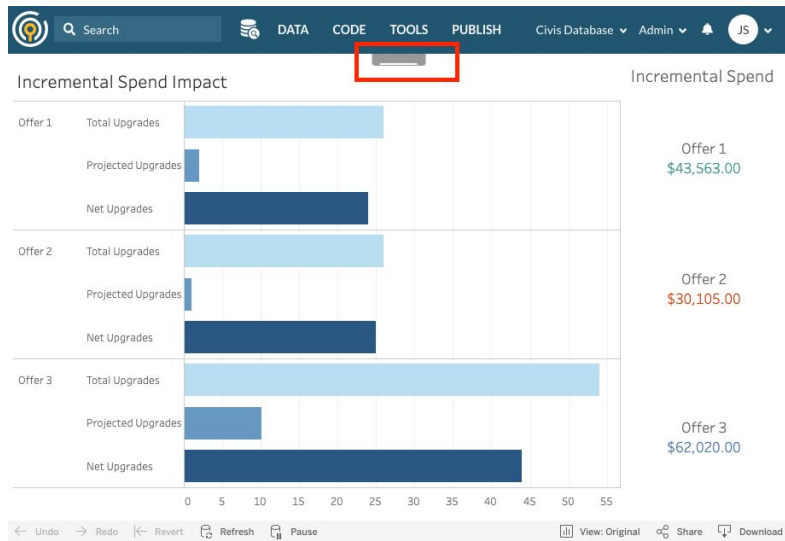
See [Reports on Civis Platform](#) for more.

Reports are Platform Objects

Sharing, Organizing and Archiving

Reports are Platform objects and obey the same sharing and permission rules as other objects.

However unlike other Platform objects they are displayed in fullscreen mode by default and users must exit fullscreen to see the action menu.



See [Reports on Civis Platform](#) for more information.

Tableau

Prerequisites and Connecting to Data

To publish Tableau reports to Civis Platform, users will need to do the following:

1. Request access to the *Civis VPN* and a set of *Tableau Server* credentials by emailing support@civisanalytics.com
2. A License to use, and a copy of [Tableau Desktop](#) - Civis currently supports Tableau Desktop versions up to 2021.3.x *only*
3. Read access on the data you wish to visualize

Connecting to Data Sources

Civis recommends that you create your data sources as extracts, rather than via live connections. Extracts compress your data so that the report does not have to query source data in real time. This improves the performance of your visualization and allows for a better report viewer experience.

See [Publishing Reports to Tableau](#), [VPN support](#), and [Granting Permission to Database Objects](#) for more



Tableau Basics

Publishing to the Civis Tableau Server Integration

- Reports published to the Civis Tableau Server are automatically created as Tableau Report objects in Civis
- To ensure your report shows up in Platform, **Show sheets as tabs** must be selected

Publish Workbook to Tableau Server

Project
Default

Name

Description

Tags
Add

Refresh Schedule (Full Extract)
None

Sheets
All Edit

Permissions
Same as project (Default) Edit

Data Sources
1 embedded in workbook Edit

More Options

Show sheets as tabs

Show selections

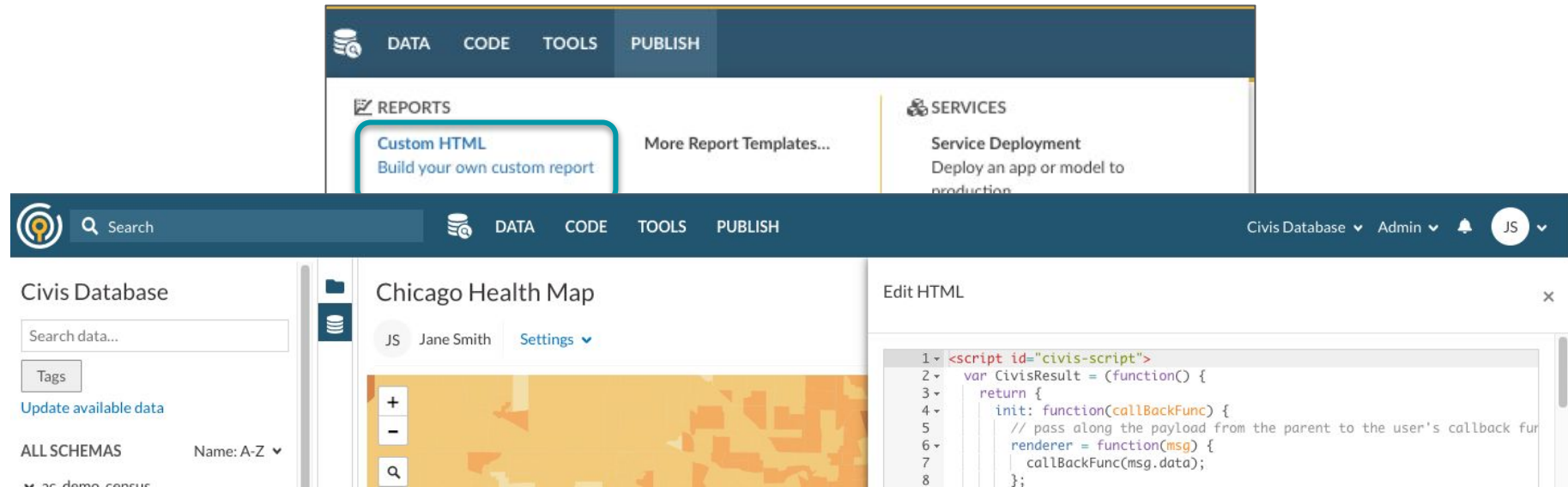
Include external files

Warning:
Server is running an older version than your copy of Tableau Desktop. The workbook will be downgraded on the server and may lose some functionality. All datasources must be embedded in the workbook.

Specify the workbook name Publish

- Non-FedRamp Clients can request Public Links of published Tableau reports by contacting support@civisanalytics.com
- See [Refreshing Tableau Reports with the Civis API](#) for more about keeping your reports up to date with the freshest data

HTML Reports



The screenshot displays the Civis platform interface. At the top, a dark blue navigation bar contains icons for DATA, CODE, TOOLS, and PUBLISH. Below this, a 'REPORTS' section is visible, with a 'Custom HTML' report template highlighted. The 'Custom HTML' template is described as 'Build your own custom report'. To the right, a 'SERVICES' section is partially visible, showing 'Service Deployment'.

The main interface is divided into three panels. The left panel, titled 'Civis Database', includes a search bar, a 'Tags' button, and a list of 'ALL SCHEMAS'. The middle panel, titled 'Chicago Health Map', shows a map visualization with a yellow and orange color scheme. The right panel, titled 'Edit HTML', displays a code editor with the following JavaScript code:

```
1 <script id="civis-script">
2   var CivisResult = (function() {
3     return {
4       init: function(callbackFunc) {
5         // pass along the payload from the parent to the user's callback func
6         renderer = function(msg) {
7           callbackFunc(msg.data);
8         };
9     };
10  }
```

Platform provides users with the ability to build and use Custom HTML reports. To make these HTML reports dynamic, you can implement the Javascript library D3 into the report's HTML. If you're inexperienced or unfamiliar with D3, head on over to its [official website](#) to learn more. Or if you'd like to dive into a bunch of tutorials, check out this helpful [GitHub repository](#).

R Markdown as a Civis Report

For R users, R Markdown is widely-used tool for displaying analyses and visualizations produced in R. Using the `publish_md()` function from the Civis Platform R API Client, users can publish [HTML documents published from R Markdown](#) as a HTML report in Civis Platform.

To publish your R Markdown report:

1. Create a blank Custom HTML report in Platform
2. Store your `.Rmd` file in a GitHub repository, along with a `.R` file containing a call to `publish_md()`. This `.R` file can be as simple as:

```
library(civis)
publish_rmd("/app/my_report.Rmd",report_id = 83263,report_name = "Platform Markdown Report")
```

3. Use a Platform container script to run your `.R` file

Civis Services

More advanced forms of reporting, including the following, can be deployed in Platform as a Civis Service:

- Shiny (R) - see [Shiny Apps in Civis Platform](#) and [Shiny App Deployment Example](#) for more information.
- Flask (Python)
- React (Javascript)

For more details on how to configure and deploy your app as a Civis Service, see [Civis Service Deployment](#) and [Advanced Services](#).

Other Reporting Methods with Civis Platform

Google Sheets

Civis Platform allows you to link your Google and Civis accounts to easily export your SQL result sets into Google Sheets.

A couple of Google Sheet best practices/reminders:

1. Google Sheets has a limit of 5 million cells per worksheet. To avoid hitting this limit, ensure you are exporting data from *reporting tables*.
2. Civis Platform does not provide you with the ability to format your Google Sheets (i.e. we do not implement the Google Sheets API to allow you to specify conditional formatting rules or spreadsheet design). As a workaround, export raw results to a worksheet and use formulas to produce a finished product

See [Export to Google Sheet](#) for information on this feature and [Creating Google Sheet Exports via Civis API](#) for information on how to create these exports programmatically.

Platform Connector for Google Data Studio

Overview

The Civis Platform Google Data Studio (GDS) Community Connector allows you to add Civis Platform as a Data Source when building reports in GDS.

Requirements and Limitations

- Requires a Civis Platform API key to connect
- Community Connector Data Sources are limited to 1 million rows
 - Civis recommends using < 100K rows for best performance
- Community Connectors limit use < 30 data sources per report

Data Freshness

Google Data Studio caches the results of queries and table exports. For community connectors, including the Civis Platform connector, the freshness options are set to 12 hours. Please see the GDS [documentation on Data Freshness](#) for more.

Platform Connector for Google Data Studio

Setup

1. In Google Data Studio, Create > Data Source (Note: this can also be done from a report: Add Data > Connect to Data)
2. Search for “Civis Platform”, it will appear under “Partner Connectors”
3. Input Platform API key
4. After authenticating, follow the instructions and fill in the fields
 - Credential ID (or leave blank to use your default credential)
 - Database Name
 - Data Source type Custom SQL or Table
 - i. SQL Statement
OR
 - ii. Schema and Table
5. Click Connect

See [Working with Google Data Studio in Platform](#) for more.

Platform Job Email Notifications

In Platform scripts, users have the ability to create JSON run outputs. Individual values from the JSON can then be used as values in the body of email notifications.

For example, say the JSON we created as a run output is: `json_value_dict = {'sales': 85}`. If you put `{{sales}}` in the email body, you will see that in the email.

On Success Off On

Email Recipients*

oharringtonwoodard@civisanalytics.com

Separate multiple emails with comma

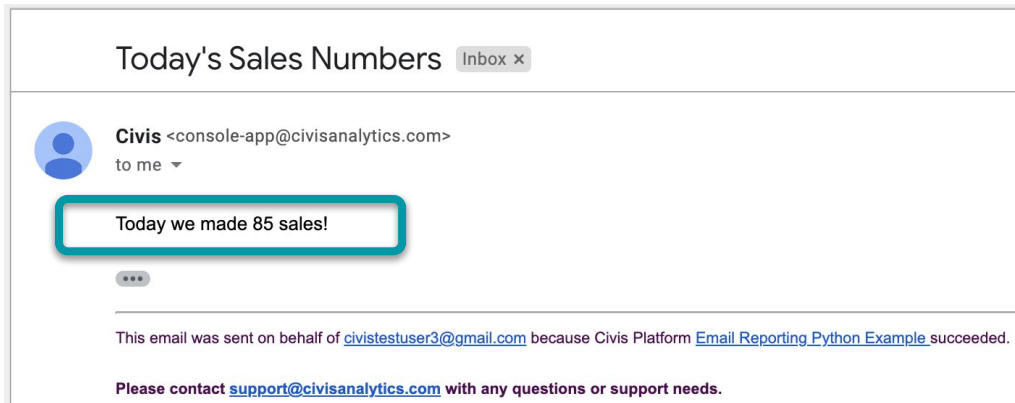
Email Subject

Today's Sales Numbers

Email Body

Today we made {{sales}} sales!

You can use [markdown](#) to add formatting



See the [Run Outputs section of The Welcome to Scripts](#) for more information on creating and using run outputs.

Email

The body of Platform notification emails is markdown compatible allowing you to easily include tables and links.

Links can be formatted in markdown syntax `[text](link)`. See more markdown tips [here](#).

If you wish to produce a table in your notification email, create a markdown table. Civis has example code for [Python](#) and [Javascript](#) on how to produce a markdown table to be displayed in notification emails. For example, if you assign your markdown table to the variable `md_table`, if `{{md_table}}` is included the notification email body, you will receive an email that looks like:

Civis <console-app@civisanalytics.com>
to me ▾

<i>PARKKEY</i>	<i>PARKNAME</i>	<i>PARKALIAS</i>	<i>CITY</i>	<i>STATE</i>	<i>COUNTRY</i>
ALB01	Riverside Park		Albany	NY	US
ALT01	Columbia Park		Altoona	PA	US
ANA01	Angel Stadium of Anaheim	Edison Field; Anaheim Stadium	Anaheim	CA	US
ARL01	Arlington Stadium		Arlington	TX	US
ARL02	Rangers Ballpark in Arlington	The Ballpark in Arlington; Ameriquest FI	Arlington	TX	US
ATL01	Atlanta-Fulton County Stadium		Atlanta	GA	US
ATL02	Turner Field		Atlanta	GA	US
ATL03	Suntrust Park		Atlanta	GA	US
BAL01	Madison Avenue Grounds		Baltimore	MD	US
BAL02	Newington Park		Baltimore	MD	US
BAL03	Oriole Park I		Baltimore	MD	US
BAL04	Belair Lot		Baltimore	MD	US
BAL05	Monumental Park		Baltimore	MD	US

Email

Notification emails can also be useful for reminding stakeholders to take some action. Since the body of Platform notification emails is markdown compatible, you can add hyperlinks! This is useful if you need others to check a recently updated report.

For example, in the Platform script that updates your Google Sheet export, include the following to hyperlink your Google Sheet report in the notification email:

Notifications

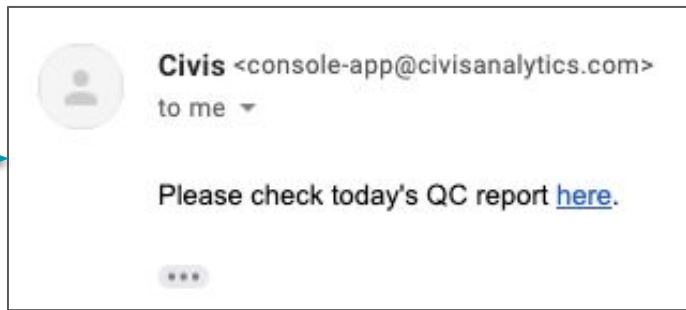
Success Notifications Off On

Recipients
comma-separated email addresses

Email Subject

Email Body
You can use **markdown** to add formatting. You can put `[link to file]` `{{file_url}}` in your body if you want to link to the file.

URLs





Thank you!

If you have any questions, please contact Civis Client Success at support@civisanalytics.com.