



ReportPlus Architecture Guide

Disclaimer

THE INFORMATION CONTAINED IN THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT ANY EXPRESS REPRESENTATIONS OF WARRANTIES. IN ADDITION, INFRAGISTICS, INC. DISCLAIMS ALL IMPLIED REPRESENTATIONS AND WARRANTIES, INCLUDING ANY WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

ReportPlus™ – Architecture Guide 1.0.

All text and figures included in this publication are the exclusive property of Infragistics, Inc., and may not be copied, reproduced, or used in any way without the express permission in writing of Infragistics, Inc. Information in this document is subject to change without notice and does not represent a commitment on the part of Infragistics, Inc. may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents except as expressly provided in any written license agreement from Infragistics, Inc.

Infragistics, Inc. and ReportPlus are trademarks of Infragistics in the United States and/or other countries.

This document also contains registered trademarks, trademarks and service marks that are owned by their respective owners. Infragistics, Inc. disclaims any responsibility for specifying marks that are owned by their respective companies or organizations.

©2016 Infragistics, Inc. All rights reserved.

Table of Contents

Introduction.....	4
Chapter 1 ReportPlus Fundamentals.....	5
Accessing your data with ReportPlus	6
Dashboard Storage.....	8
Complete Component Summary	11
Chapter 2 Data Layers and Architecture.....	12
Data Layer and Connectors	13
ReportPlus Clients Architecture	15
ReportPlus Server Architecture	17
Appendices.....	19
Appendix 1: Document Changelog	20

Introduction

Welcome to the ReportPlus Architecture Guide.

The purpose of this guide is to build a knowledge base for help developers and solution architects to help them build effective quality applications using the ReportPlus platforms; it provides guidance information on every specific structure and explains the application behavior in easy-to-understand flow diagrams. The guide is presented in sections that correspond to major architecture and design focus points. It is intended to be used as a reference resource.

About ReportPlus

ReportPlus is a dashboard-authoring tool that allows you to create rich visualizations retrieving data from multiple data sources like relational databases, cloud services and others.

You can use ReportPlus in multiple ways and in multiple platforms, as a stand-alone application in a single device, as a personal application across all of your devices or as part of an enterprise deployment. The connections to the data sources, in all cases, is established directly from the device itself. There is no need for a server side component to use ReportPlus.

Using ReportPlus on your devices

The easiest way to use ReportPlus is to download the iOS-based application from the [Apple AppStore](#), and start creating your own dashboards. You can also get a Desktop version for Windows from Infragistics Website.

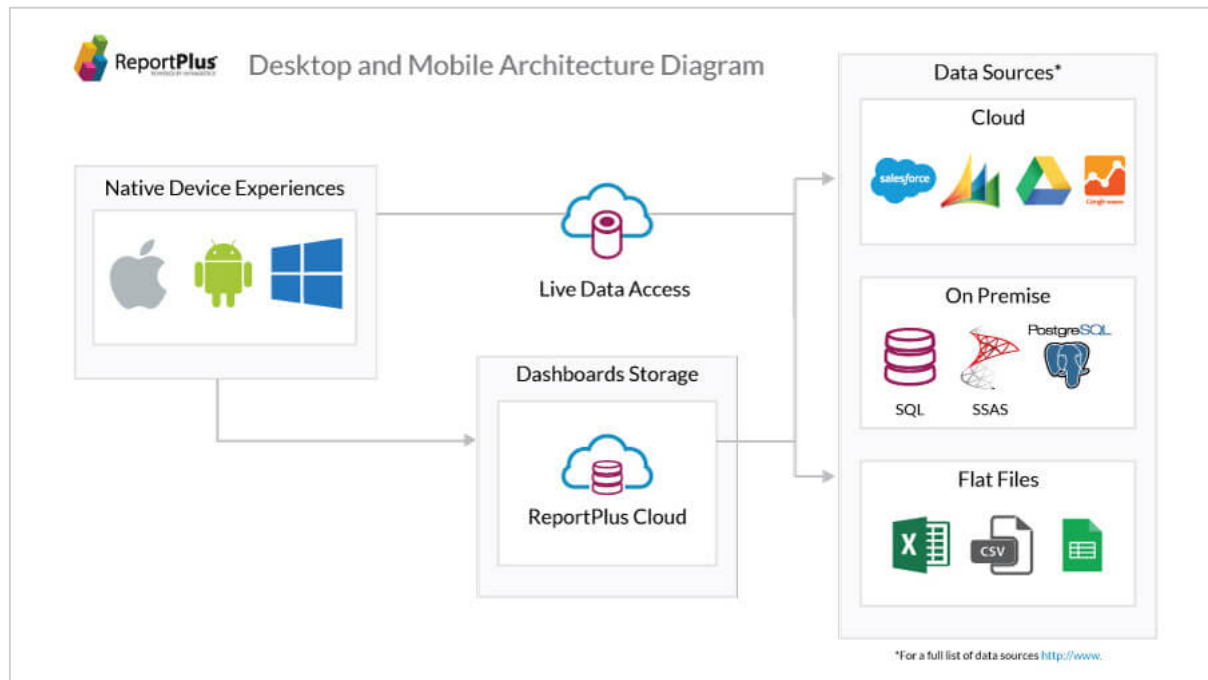
Once you create your dashboards, you can view them in other platforms. For instance, you can use ReportPlus for Android to open dashboards created in other platforms.

Chapter 1

ReportPlus Fundamentals

Accessing your data with ReportPlus

The following diagram shows how ReportPlus connects to data sources, and how it supports multiple repositories to store the dashboards.



ReportPlus iOS, Android and Desktop

The iOS, Android and Desktop platforms **access the data directly from the data sources**. This is because your connection is either one of these:

- An internal local area network.
- A VPN connection.
- Cloud-service services accessible through the internet.

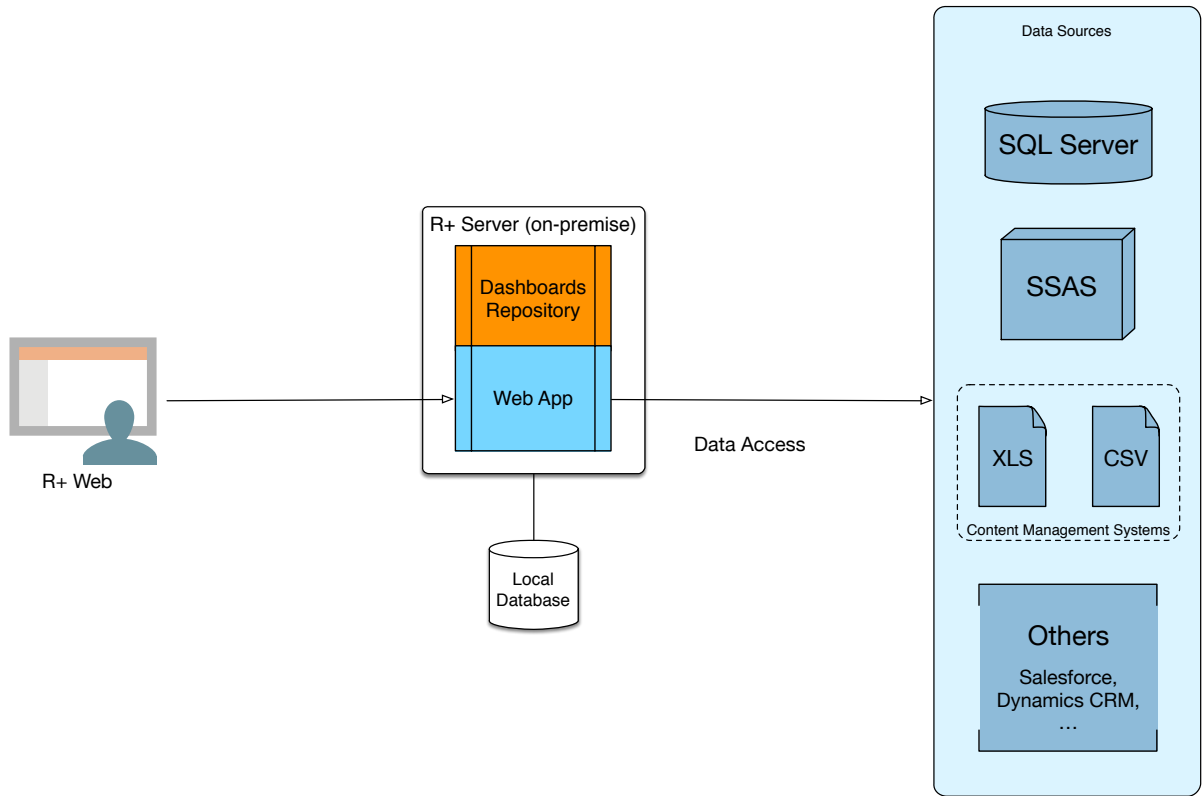
In any case, there is no need for a server-side component to be installed in advance for you to use ReportPlus in your device.

ReportPlus Server

Using ReportPlus Server means that you can open your dashboards using a standard browser application (e.g.: Google Chrome, Mozilla Firefox, Internet Explorer). In this case, you need the on-premise ReportPlus Server installed in your company.

In the case of an on-premise ReportPlus Server installation, the data is accessed by the server and sent to the Web Browser in order to be rendered.

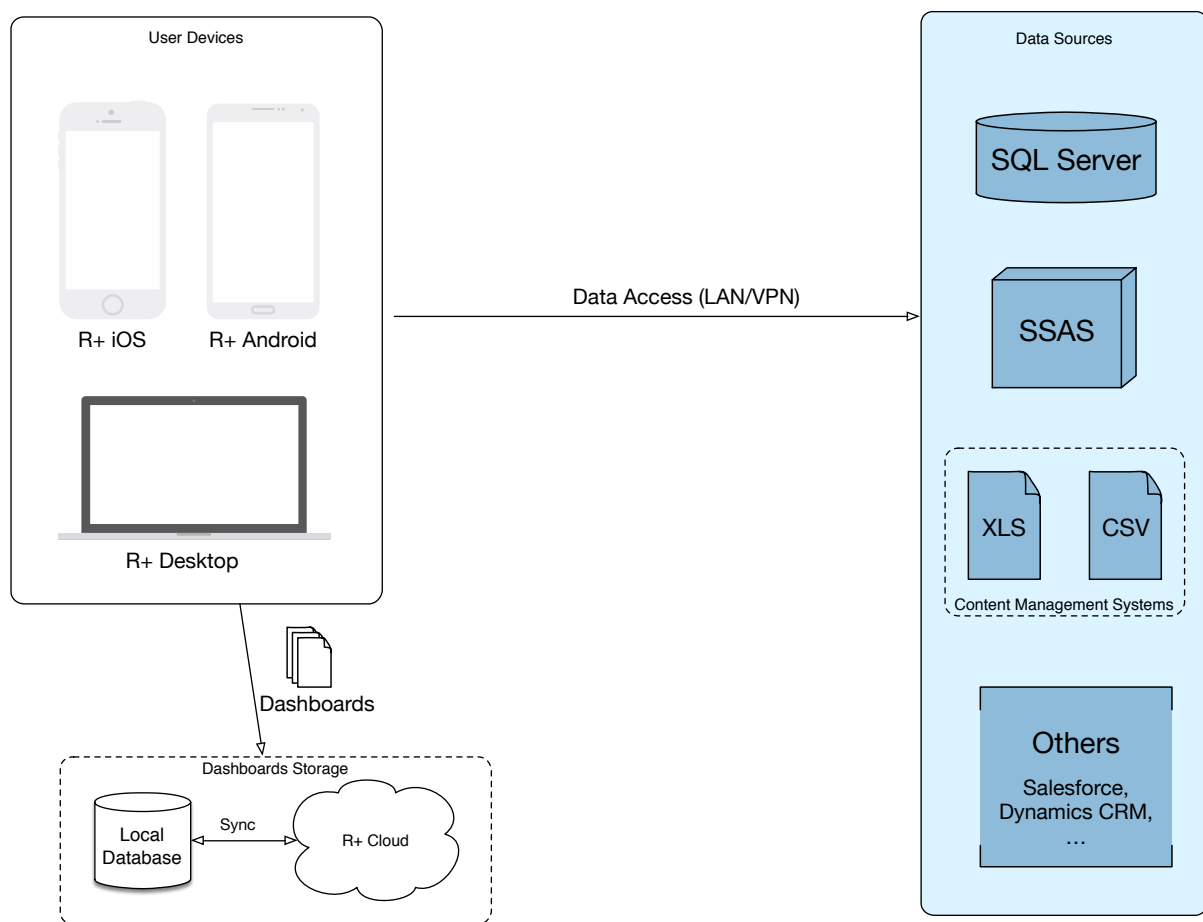
The following diagram shows how an on-premise ReportPlus Server connects to data sources:



Dashboard Storage

ReportPlus offers multiple options to store your dashboards. The simpler one is to store them in your local device, like your iPhone on your desktop PC. However, you probably want to have the same dashboard across all of your devices; if you work in a team, you might also want to share your dashboards with your colleagues.

The ReportPlus Cloud Storage provides a way to share your dashboards automatically across all of your devices. Every dashboard created on your device will be automatically synchronized with the cloud storage, and then sent to any other device associated to your account.



In an Enterprise environment, you may want to install an on-premise server to store the dashboards in your own infrastructure. In that case, you will be able to use your Active Directory to verify credentials and assign permissions to dashboards and folders, as you do in the file system.

Sharing your dashboards

There are multiple ways of sharing your dashboards with someone else, including the following:

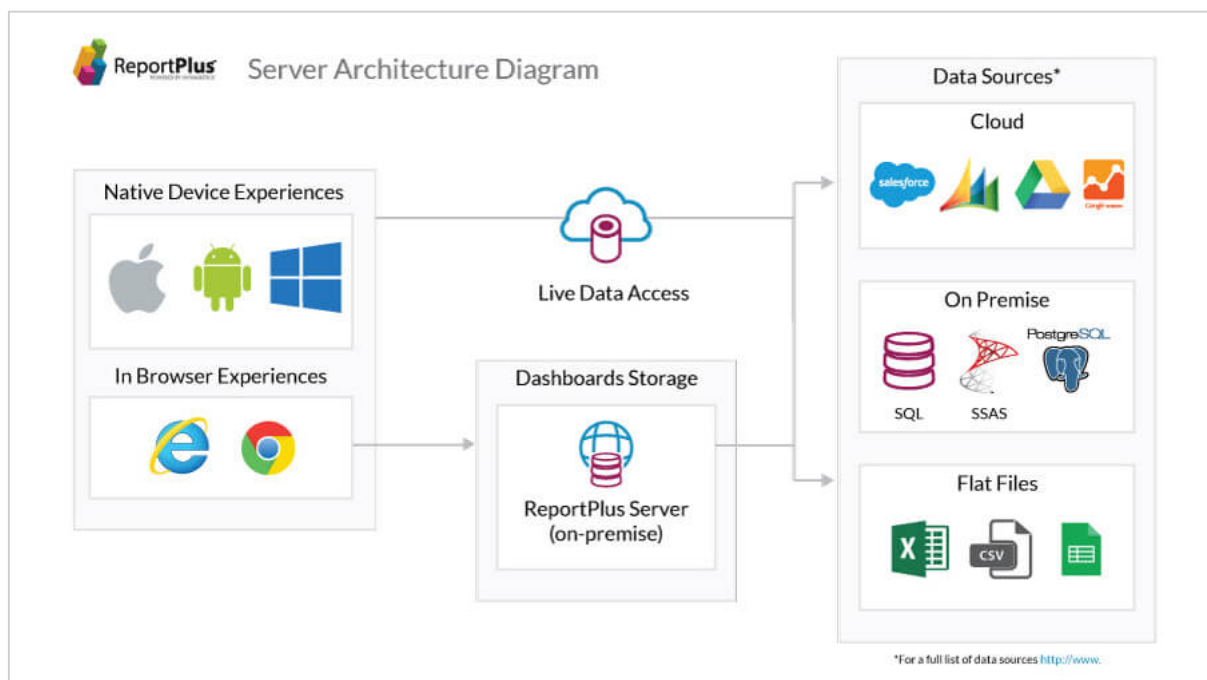
- **Using the Share feature in the Cloud Storage**, by sharing the dashboard with your colleagues.
- **Storing the dashboards in the on-premise server**, if installed in your company.
- **Sending the dashboard by e-mail** to any other user with ReportPlus installed.

ReportPlus on the Enterprise

The on-premise ReportPlus Server offers you the ability to have all the dashboards administered and stored securely in your own infrastructure. You can take advantage of your existing Active Directory structure to define who can create and access dashboards in the same hierarchical way you can assign permissions in a shared file system.

All of your devices with ReportPlus installed will access these centralized dashboards; every user with enough permissions will automatically have access to your changes. Every dashboard will be versioned, and you will be able to rollback your changes at any time by restoring a previous version of a given dashboard.

ReportPlus Server enables also the ReportPlus Web Viewer component in order for you to access all your dashboards with a standard Web Browser without the need of pre-installing software in your device.



Data security

To improve performance, ReportPlus stores cached data in the device in order to minimize the number of queries sent to the server or database. The credentials to access data sources are also stored in the device as they are requested only when the data source is configured or used for the first time. This means that sensitive information is stored in the device. ReportPlus, therefore, takes advantage of the security capabilities available on each device.

iOS

In iOS devices, the passwords are stored in the device's Keychain; if a passcode is set in the app, that passcode is used also as part of the key used for encryption as a way to have an additional security level. As long as the passcode is set on the device, the cached data and the local database is stored using iOS Data Protection, which means all the files will be encrypted as soon as the device is locked. This feature is provided by iOS itself and ensures the best performance.

Desktop and Android

In Desktop and Android, all the passwords are encrypted before being stored. As for cached data files, they are not encrypted as there is no resource similar to iOS' "Data Protection". For these platforms, we recommend using MDM platforms, which will help keep data secure.

MDM (Mobile Device Management) Platforms

Many enterprises today use MDM (Mobile Device Management) platforms, which allow your enterprise to centralize the deployment of applications for your mobile devices. It also manages the administration of all devices in your company by enforcing security policies, keeping the communication secure, the files encrypted, etc.

You can use any of the MDMs in the market for deploying ReportPlus iOS and Android to the mobile devices in your company.

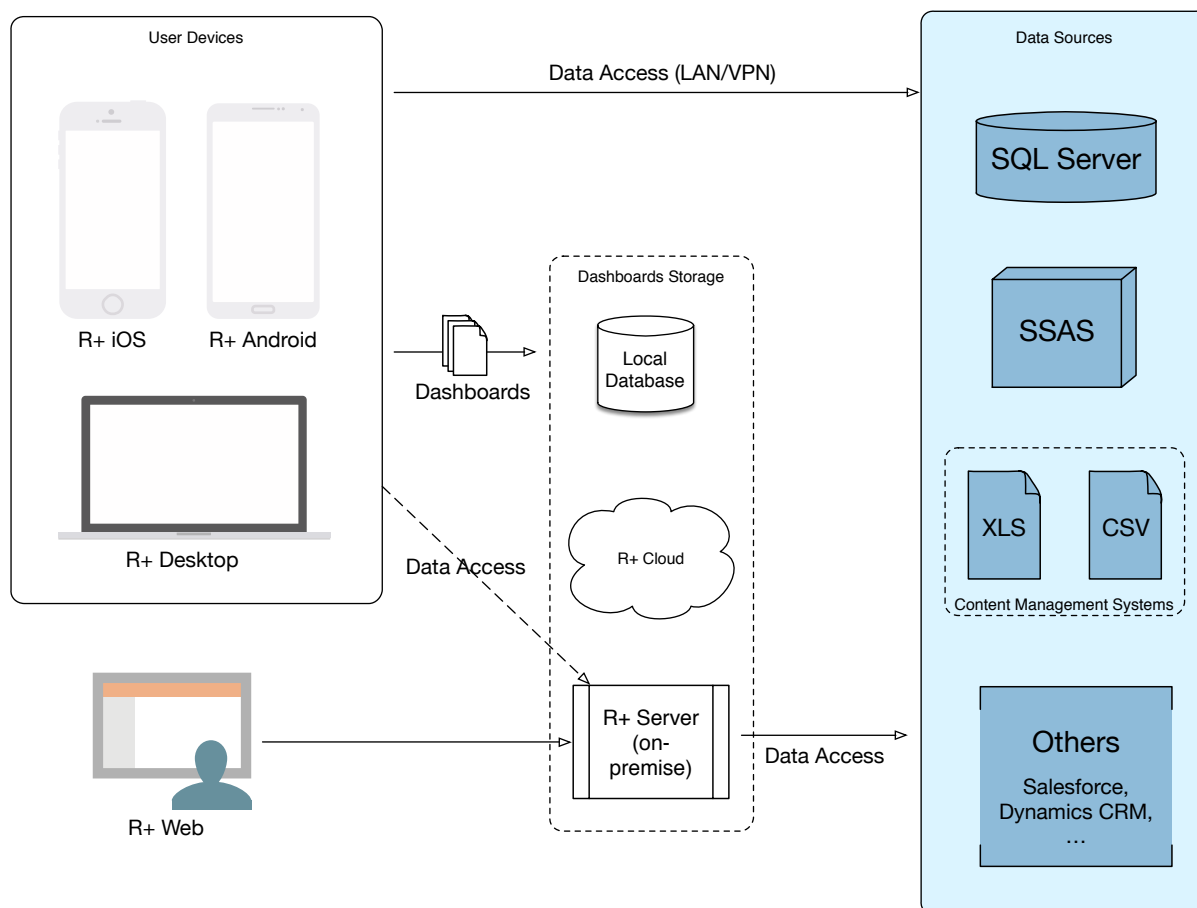
ReportPlus currently supports some additional features for MobileIron. This MDM can be used to:

- Centrally modify the configuration file for ReportPlus.
- Define policies, such as the clipboard access.
- Secure the communication channel to the data sources by using per-app-VPN in iOS devices.

Additional support for other MDM providers will be added soon.

Complete Component Summary

In the following diagram, you will see the overall structure of ReportPlus with all the components mentioned in previous sections.



As you can see, both the mobile devices and ReportPlus Server access the data directly. There also are multiple options for the storage of the dashboards definitions.

Dashboards are accessed by both the devices and the server component through these repositories. The dashboard XML contains the definition of the widgets and the data sources used, but does not contain credentials or any other kind of authentication information. Once the dashboard is opened, a Data Layer component in the device or server converts that definition to queries (SQL, MDX or others, depending on the data source) to retrieve the data from the origin.

Chapter 2

Data Layers and Architecture

Data Layer and Connectors

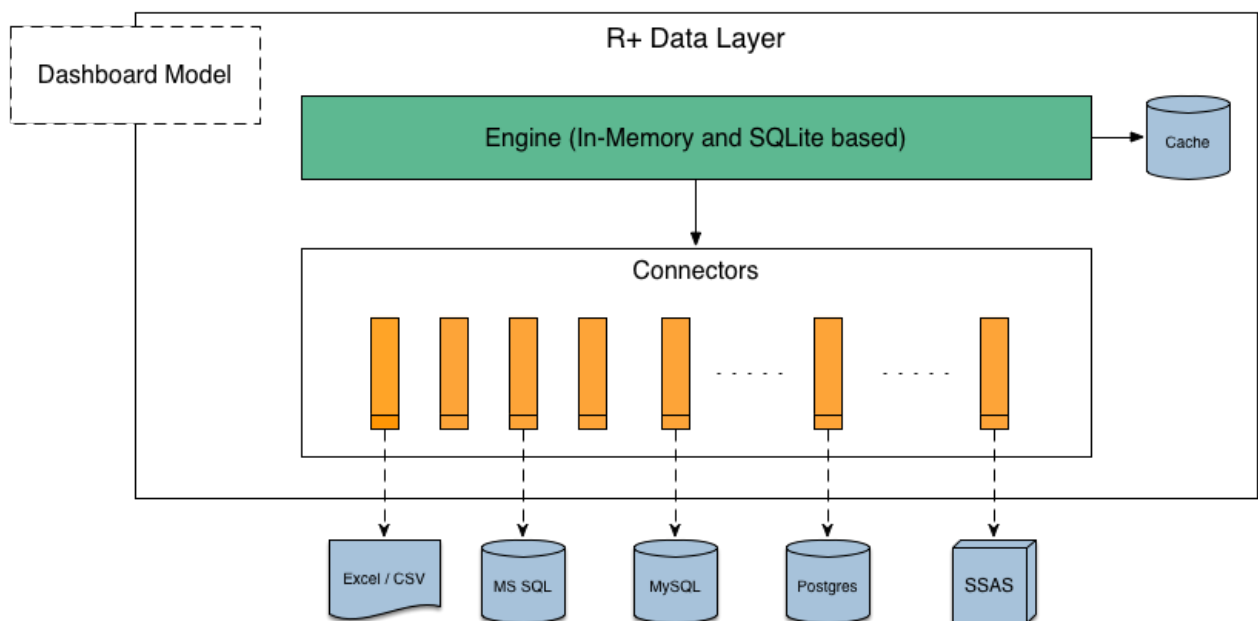
The Data Layer component provides the data to be rendered to the visualizations. It is responsible for the interaction with the connectors (that retrieve the data from each particular data source) and the post-process.

Every data source provides different capabilities in terms of data management. The below are some examples:

- **Data from CSV or Excel files cannot be filtered by the data source itself.** The data is not strongly typed, and, in the case of Excel, it might not be structured at all.
- **Data from other sources can sometimes be filtered and sorted in the data source.** This is the case, for example, with relational databases. The data is structured, since you always get the data from a table, and strongly typed, since for each column you know exactly what's the datatype and you know the database will enforce it.

The Data Layer component handles all of these cases. For CSV and Excel files, it provides data type detection. In the case of Excel files, it also provides detection for data tables inside a single sheet. For query based providers (SQL based or MDX based in the case of MS Analysis Services), it generates the query based on the requirements in the dashboard trying to leverage all of the capabilities of the given data source.

Some of these tasks are resolved by the Engine component, and some other by the specific connector for the data source. The Engine component caches data in order to improve performance and minimize the access to data sources. It also resolves some other post-processing tasks like pivoting the data, resolving calculated fields and data management operations (filters and sorting) not supported by the specific data source being used.



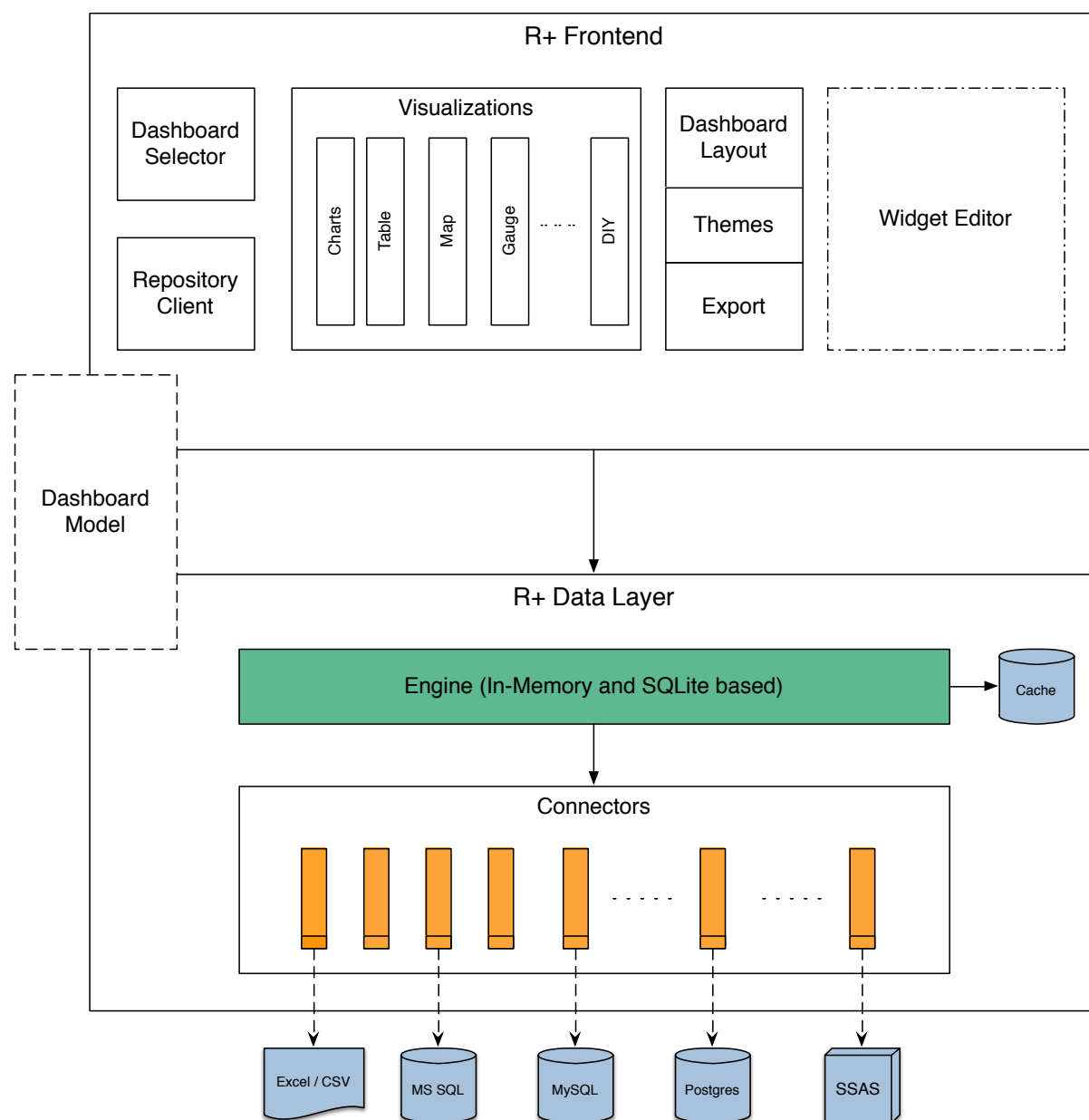
Indeed, the engine is a dual implementation running completely in memory for small amounts of data to get the best performance and using an embedded database when the amount of data cannot be handled in memory. All of the thresholds involved when deciding which implementation to use (in-memory vs embedded database) are completely configurable by administrators.

One of the more powerful features in the product allows you to join (or merge) data from different data sources. For instance, **you can display** Sales figures loaded from an Analysis Services cube or Salesforce along with Budget numbers loaded from an Excel Spreadsheet stored in your SharePoint serve, Google Drive, or others, **in the same widget**.

This data merging is processed completely by the data layer in a transparent way, using the embedded engine and then enabling data management operations like calculated fields on top of the merged data.

ReportPlus Clients Architecture

The following diagram shows the general architecture followed by ReportPlus for all platforms:



On top of the Data Layer component, the Frontend is responsible for displaying the data returned by the connectors and processed later by the Data Layer. The dashboard model contains the configuration on how to use the data to populate each of the visualizations supported.

ReportPlus Server

For ReportPlus Server, the communication between Frontend and Data Layer goes through ReportPlus Server, which exposes the required services to obtain the data client side.

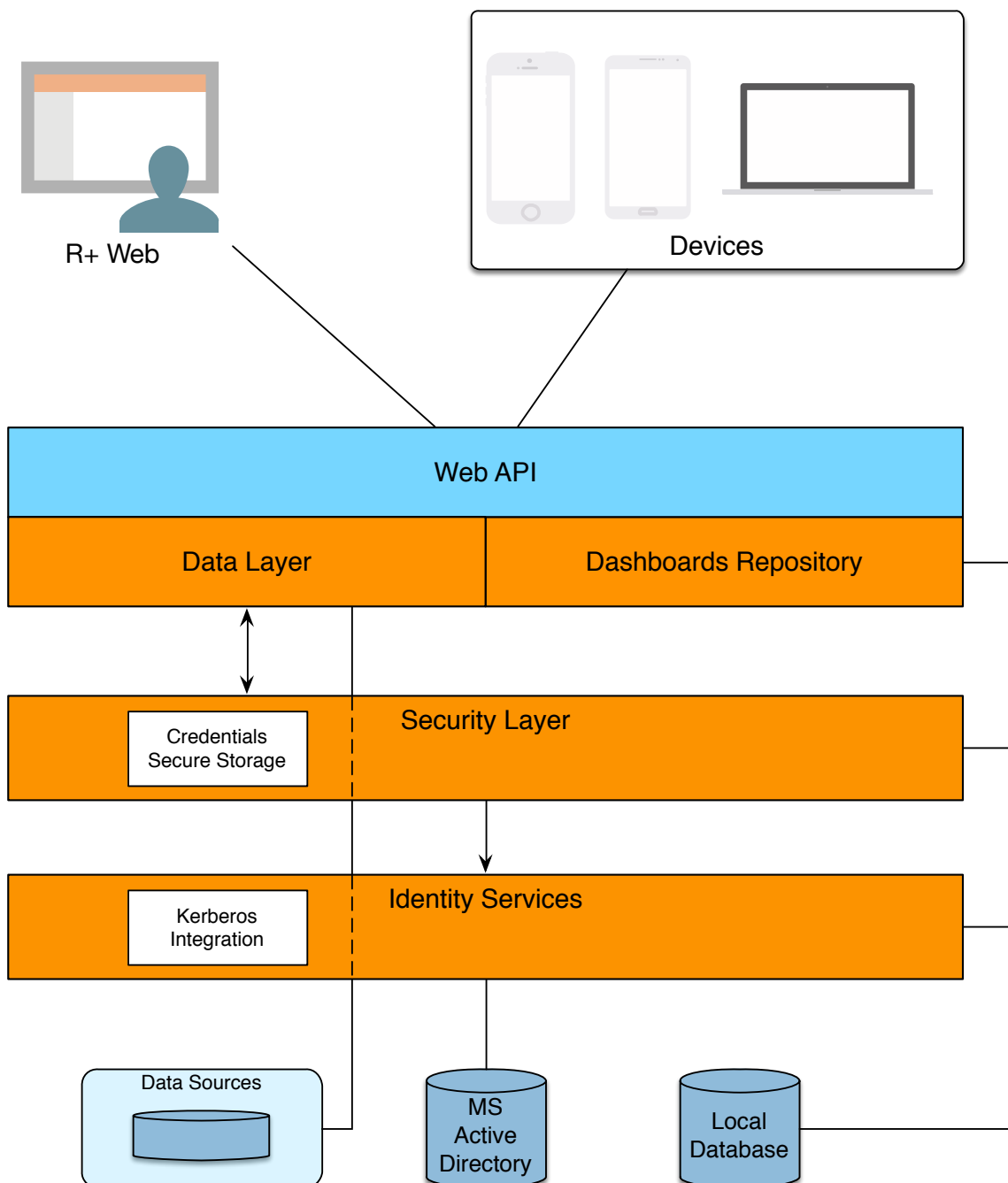
ReportPlus Desktop, iOS and Android

For ReportPlus Desktop, iOS and Android, both Frontend and Data Layer are executed inside the same process and the communication is just in-process.

ReportPlus Server Architecture

The following diagram shows the architecture for the ReportPlus Server. This server side component exposes a web API which provides the following services:

- For the Web Client:
 - Dashboard Repository administration and browsing.
 - Security administration.
 - Data access.
- ReportPlus Clients running on devices
 - Dashboard storage services.



Identity Services Component

The “Identity Services” component provides the list of users and roles to be used (by default from Active Directory) when authenticating and setting permissions. It also provides the integration with Kerberos.

This component can be replaced by a custom identity management component in order to authenticate your users using your own authentication system in the case that Active Directory is not being used.

Integration with Kerberos

The integration with Kerberos allows users to access data stored in enterprise databases (like MS SQL Server) using their domain authentication without additional authentication to the database server itself.

Security Layer

The Security Layer handles the secure storage of credentials used to access data sources and the Repository handles the storage of dashboards, their organization in folders, and the permissions associated to them.

Data Layer

The Data Layer (and its connectors) uses the Security Layer and the Identity Services component in order to authenticate to each data source. The authentication will be performed either by using explicit credentials loaded from the secure storage, or through impersonation using the Kerberos infrastructure.

Appendices

Appendix 1: Document Changelog

Appendix 1: Document Changelog

Version	Chapter	Section	Description
1.0	All chapters	All sections	Document creation.