Qlik Sense® Enterprise architecture and scalability

June, 2017

Platform

Qlik Sense® is an analytics platform powered by an associative, in-memory analytics engine. Based on users' selections, calculations are computed at runtime against data stored in-memory. Results are returned to users via a zero-footprint web interface delivered on desktops, laptops, mobile devices, and through embedded analytics. Qlik Sense offers a highly interactive, associative experience in which users can freely navigate through data with little to no constraint in their analysis path.

Associative, in-memory applications

Qlik's QIX Engine uses an associative, in-memory technology that enables users to analyze and freely navigate data in an intuitive way. This second generation of Qlik's proven data indexing engine allows users to freely explore data and create visualizations based on data from multiple data sources simultaneously, ranging from Excel and Access, to databases such as Oracle and SQL Server, to big data sources such as Cloudera and Redshift.

Qlik Sense uses columnar, in-memory storage. Unique entries are only stored once in-memory and relationships among data elements are represented as pointers. This allows for significant data compression, more data in RAM, and faster response times for users.

There are some big data scenarios where data should remain at the source. Using a built-in technique called On-Demand Application Generation, data sources can be queried based on users' selections, yet still provide an associative experience to the user.

What is Associative?

Associative refers to the unique combination of in-memory data storage and a patented engine. This allows users to search and navigate through and create visualizations upon data with a *continuous context*. This removes the constraints imposed by traditional hierarchical or query based approaches.

Qlik Sense Clients

Hub

The Hub is the Qlik Sense user environment and is delivered through a zero-footprint web browser interface. Through the Hub, all aspects of development, drag-and-drop content creation, and consumption is possible. Qlik Sense delivers a *responsive design* methodology to automatically display and resize visualizations with the appropriate layout and information to fit the device, whether it is a browser on a laptop or desktop, tablet, or smartphone. Built with current standards of HTML5, CSS3, JavaScript, and web sockets, Qlik Sense enables you build on any platform and deploy to any platform.
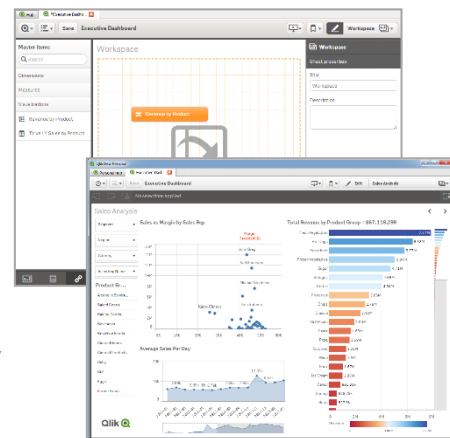
Management Console

The Qlik Sense Management Console allows administrators to govern all aspects of the Qlik Sense platform, ranging from data connectivity, application and task management, to security administration, monitoring, and auditing.
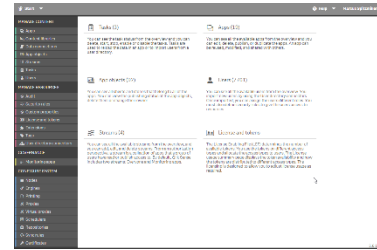
Embedded Analytics

Qlik Sense APIs are REST and JSON based and are the same APIs that were used to develop Qlik Sense itself. You can extend Qlik Sense by adding your own visualizations and your own data sources. You can even integrate Qlik Sense visualizations and data with other web based applications. Qlik Sense provides a built-in web-based editor for working with these APIs.
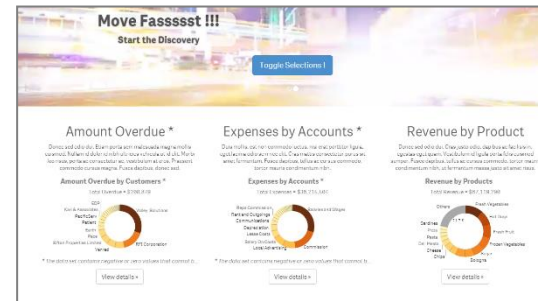
See the data sheet titled Qlik Sense APIs for more information.

Architectural Overview

Qlik Sense is designed to support visual analytics in a scalable, secure, and governable way. Qlik Sense can be deployed on a single server and scale vertically and horizontally to address the availability and processing requirements of your deployment, whether on premise or in the cloud.
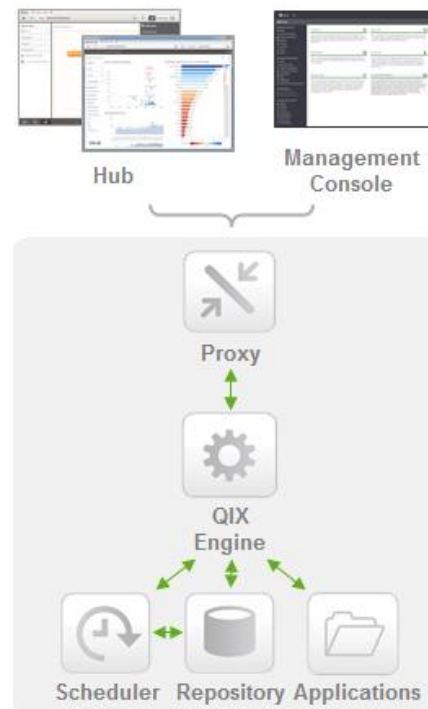
- **Qlik Sense Proxy (QPS)** is the entry point for users and administrators via the Hub and Management Console, respectively. This component integrates with many identity providers. (e.g., Active Directory, SAML, etc) The Proxy also manages sessions, provisions licenses, and handles load balancing to the other components.

- **Qlik Sense Engine (QIX)** is the associative, in-memory data indexing engine. This RAM-based application tier delivers highly interactive self-service visualizations, search, and calculations at runtime.

- **Qlik Sense Scheduler (QSS)** is the scheduling component that coordinates data loads. This scheduler supports time based application reloads, dependancies, and may be driven by enterprise schedulers (e.g., CA AutoSys).

- **Qlik Sense Repository (QRS)** is the centralized store of configuration and management information of the platform. This component manages user definitions, security, and many other elements of the platform. The data persists in PostGreSQL.

- **Qlik Sense Applications (.QVF)** are comprised of highly compressed data, a data model, and presentation layer. These applications are persistently stored on a file system and are loaded into memory by the Qlik Sense Engine as users request them.

  Ancillary Services (not pictured): Qlik Sense Printing Service (QPR) is used for content exporting and **Qlik Sense Service Dispatcher (QSD)** calls routines for data preparation, application migration, and on-demand app generation.

Multi-Node Architecture

Qlik Sense has a modular architecture designed to meet the unique requirements of your organization. While a single Qlik Sense server can often support hundreds of concurrent users, Qlik Sense can scale even further with its multi-node architecture.

See the technical brief titled [Qlik Sense Performance Benchmark](#) for more information.

Distributed Workloads

The Qlik Sense components can be deployed on servers in different combinations to accomplish specific functions. The table below describes how the components of Qlik Sense are deployed to allow servers to function as Consumer nodes, Developer nodes, Scheduler nodes, and the Central node.

| Function	Description	Component						Comment
		Proxy	Engine	Scheduler	Repository Service	Repository Database	Application Filesystem	
Consumer Node	Consumption and Self-Service Analytics for Users	X	X		X			Proxy can be separated to its own server.
Developer Node	Create and reload new applications for users	X	X		X			For business users. Isolated non-production environments recommended for IT oriented authors.
Scheduler Node	Application and Data Reloads		X	X	X			Engine performs reloads invoked by scheduler.
Central Node	Master Scheduler, Application Upgrades, misc.	X	X	X	X	X	X	Can be co-located with other nodes.

Central Node

Every Qlik Sense environment has a Central Node that acts the master scheduler that coordinates application reloads, performs application upgrades when the software is updated, and serves extension objects to other nodes. By default, the Central Node also holds the Repository Database and Application Filesystem. In a multi-node environment, each Repository Service communicates with the Repository Database on the Central Node, and each Engine loads applications (.QVF) from the Application Filesystem on the Central Node. Optionally, the Repository Database and/or Application Filesystem can be moved to dedicated, resilient resources. Finally, because of its limited responsibility the Central Node can co-locate on another node, most commonly the scheduler(s). The Central Node functions can be nominated to run on one or more nodes, and in the event of a failure one of the candidate nodes will assume Central Node responsibilities.
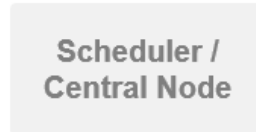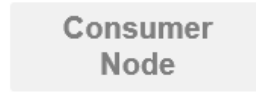
Single Node Deployment

This single-node deployment example co-locates all Sense component on a single node.

Consumer / Scheduler / Central Node
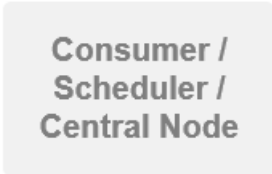
Two Node Deployment
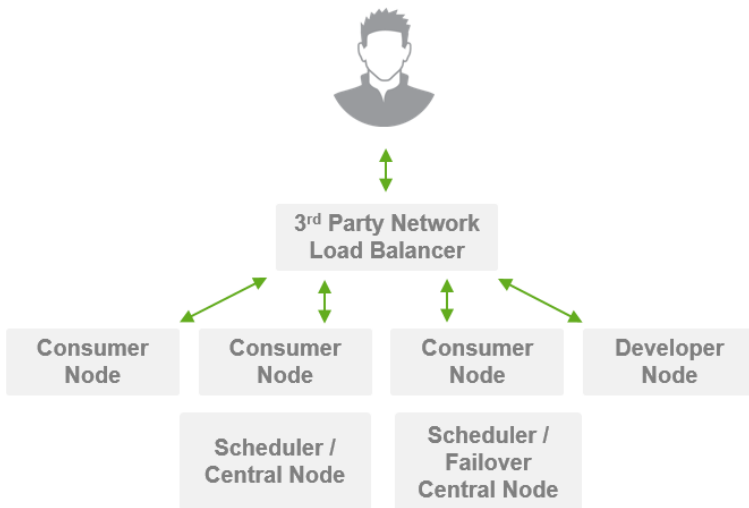
This two-node deployment example separates the workload generated by users performing analysis from data loads managed by the scheduler.

Consumer Node

Scheduler / Central Node

Scaling horizontally – Multiple Nodes

Qlik Sense can scale workloads vertically and horizontally. Within a single site (environment), multiple Consumer nodes and Developer nodes for self-service analytics and Scheduler nodes for data loads enable Qlik Sense to scale horizontally for resilience and a growing workload.

3rd Party Network Load Balancer

Consumer Node

Consumer Node

Consumer Node

Developer Node

Scheduler / Central Node

Scheduler / Failover Central Node

About Workloads

Security Rules are configurable and further enable you to dedicate workloads to specific nodes. For example, within **Consumer and Developer nodes**, isolate content creation from consumption or segment applications by business unit. Within **Scheduler nodes**, isolate data layer creation from application creation, or segment data loads by complexity or duration.

This completely configurable capability is discussed further in the document titled "Qlik Security Overview Whitepaper"

Qlik Sense APIs

Many organizations are tasked with integrating their platforms into an ever more complex and changing enterprise architecture. The Qlik Sense APIs ensure that Qlik Sense can integrate seamlessly into your environment to provide the security, automation, and delivery capabilities that are required.

The Qlik Sense APIs are REST and JSON based and are the same APIs used to develop Qlik Sense itself. This gives you tremendous power to build your own applications on the Qlik Sense platform. You can extend Qlik Sense by adding your own visualizations and your own data sources. You can leverage your existing security providers. You can integrate Qlik Sense with web applications or with desktop applications based on the .NET framework. You can manage licensing, users, and tasks programmatically.

Qlik Sense Extensions API
- Create New Visualizations

Qlik Sense Capability API / .NET SDK
- Embed Visualizations and Functionality

Qlik Sense QVX SDK
- Create Custom Data Connectors

Qlik Sense Repository Service API
- Management and Administration

Qlik Sense Proxy Service API
- Authentication, Load Balancing, Session Management

Qlik Sense User Directory Connector API
- Create Custom User Directory Connectors

Qlik Sense Engine / Backend API
- Communicate Directly with Associative Engine

See the data sheet titled "Qlik Sense APIs" for more information.

Scalability at a glance

As the needs increase to deploy a system to larger numbers of users containing larger volumes of data and larger numbers of applications, scalability becomes increasingly important. Scalability is defined and measured along several dimensions and are categorized into five main areas: architecture, data, applications, users, and delivery.

Through these areas we will discuss Qlik Sense's ability to scale.

Architecture

Qlik Sense is comprised of five modular primary services that can be scaled vertically and horizontally to meet demand. This simple architecture ensures predictability in performance and scaling.

Data

Qlik Sense scales linearly with data volumes. Response times for analysis and ETL from multiple data sources are predictable as data volumes grow.

Applications

Qlik Sense is predictable in how resource consumption scales with applications. The resource requirements of each individual application can be added to understand the requirements of a group of applications.

Users

Qlik Sense scales linearly with users. As the number of content creators and consumers grow, their response times and impact on server resources grows predictably and linearly.

Delivery

Qlik Sense enables your organization to align the work with the people most suited to do it. IT can efficiently manage hardware and secure and govern data, while users create content, self-serve, and collaborate. Fewer resources can empower many.

Qlik Sense Scalability

Architecture: In-memory, distributed, and performance and capacity scales linearly with processor cores and servers.

Data: CPU, RAM, and response times scale linearly with data.

Applications: performance is additive, ensuring predictability.

Users: CPU, RAM, and response times scale linearly with users.

Delivery: fewer administrative resources and self-service content creation allows Qlik Sense to scale with your organization.

Scalable architecture

QIX Engine

The QIX Engine is an associative, in-memory data indexing engine. This RAM-based application tier delivers highly interactive self-service visualizations, search, and calculations at runtime.

Qlik Sense calculates aggregations at the time it is requested. For example, when the user interface requires aggregates to show a chart object or to recalculate when selection is made, the aggregation is done at the same time. The QIX engine is a 64-bit process, is multi-threaded, and is optimized to take advantage of all available processor cores. As you will see, available cores will be used linearly when performing calculations for growing data volumes and users.

Memory and Cache Management

Qlik Sense uses RAM efficiently and in three ways. Qlik Sense uses RAM to store:

- The dataset that is defined by the application data model
- The cached results that are a result of engine calculations and user selections
- The session state for each user

Qlik Sense applications persist on disk on a server, and when a user requests an application it is loaded into RAM if it had not been loaded by a previous user. Multiple users leverage this application in RAM; it is loaded a single time and is not duplicated for multiple users accessing and analyzing it. RAM is freed when the application is no longer used.

As users make selections within applications, the QIX Engine performs calculations in real time. During a calculation, RAM may be temporarily allocated to perform the calculation. The results are stored in a Result Cache and are reused if any user makes the same selection (provided security allows for the same data visibility). This simple caching mechanism frees the CPU from doing additional, duplicate work and is automatically managed by Qlik Sense, requiring no administration.

Finally, each user uses a small amount of RAM to maintain session state information.

Memory Management

Applications
Highly compressed data; columnar storage; single copy of application data in RAM

Result Cache
Stores calculation results; automatically reused; self-managing and never out of sync with application data

Distributed Architecture

Qlik Sense has a modular architecture that can be deployed to multiple nodes and is supported on physical hardware, virtual platforms, and cloud environments.

With this multi-node architecture, Qlik Sense can scale linearly when additional QIX Engines are added to a deployment. Qlik Sense can achieve this scalability given it is an in-memory analytics platform, and as such does not suffer common bottlenecks prevalent in query based tools that rely on an underlying database to perform the processing. When a deployment is scaled out by adding a node and doubling the processing capacity, it can handle twice the number of users assuming a typical user produces similar amount of load. Additionally, with Qlik Sense's ability to define which applications reside on a node even better effects can be achieved. Of course, many factors contribute to scalability and overall design of a deployment. See "Deploying multi-node sites" on help.qlik.com or contact Qlik with more questions.
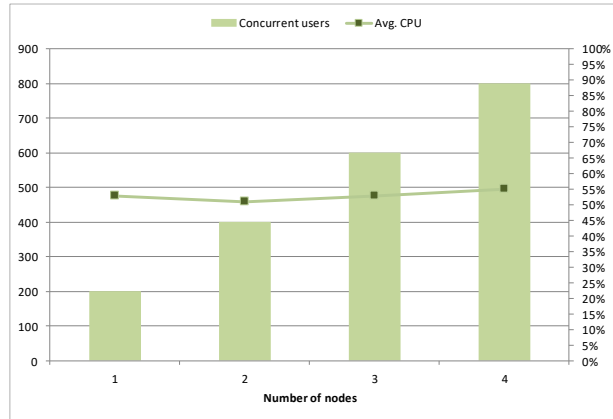
Scalable with applications and data

Qlik Sense deployments commonly grow to address the needs of many users across many business units. This growth requires a scalable solution in terms of data and application management. Sense is predictable in what impact an addition of applications or data will have on resource consumption. Resource characteristics (CPU and RAM consumption) for single applications simply add up when deployed together allowing the ability to estimate the total needs prior to publishing to production environment. The figures to the right show how resource utilization sums predictably. When three different applications have been utilized independently on their own servers their total RAM and CPU consumption sums to when the same scenario has been served in parallel at the single server. This allows for confidence when capacity planning and predictability while deploying.
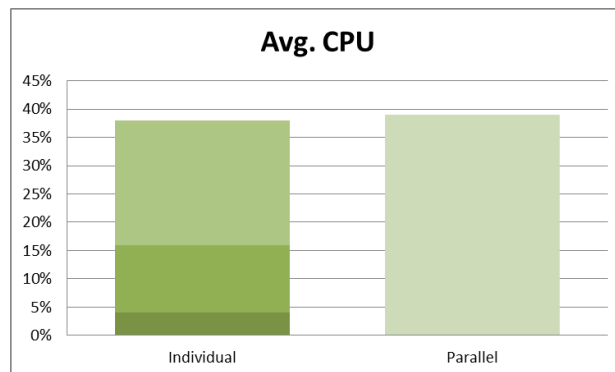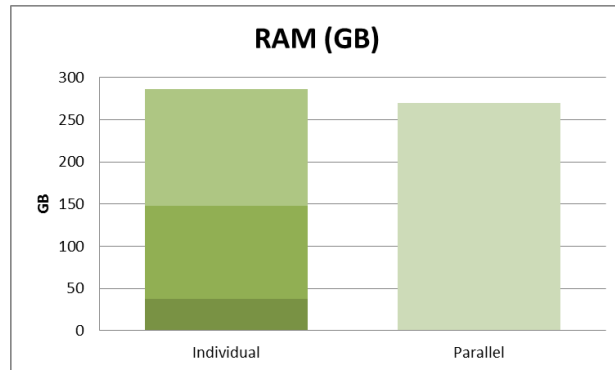
Application performance

As an application's source data volume grows, it requires a proportional amount of RAM to hold it. This does not suggest, however, that twice the source data requires twice the RAM. Qlik Sense uses a highly efficient algorithm to store each unique entry once and use pointers to reference the relationships among data. This ensures that as additional data is added, it is represented in RAM as efficiently as possible. Depending on the nature of the data being compressed, then, source data is often compressed in RAM by a factor of 10.

As an application's source data volume grows, it also requires a proportional number of CPU cycles to process the same calculation. Provided a result isn't already in the Result Cache, CPU cycles are used whenever a new aggregation is requested, and the amount of time needed to respond to an end-user request is a function of the CPU's ability to perform a recalculation on the data. Therefore, as more data is added to an application, the CPU will be required to perform recalculations over larger slices of data, resulting in more required CPU cycles. Fortunately, for the same reason cited above, Qlik Sense compression ensures that twice the data does not require twice the CPU power, as shown above.

Qlik Sense's performance scales uniformly with data. As more data is added to a Qlik Sense application, then a corresponding addition of a combination of RAM and CPU capacity allows end-user performance to be maintained in a predictable fashion. The linear scaling of CPU and RAM ensures predictably of performance and capacity planning.
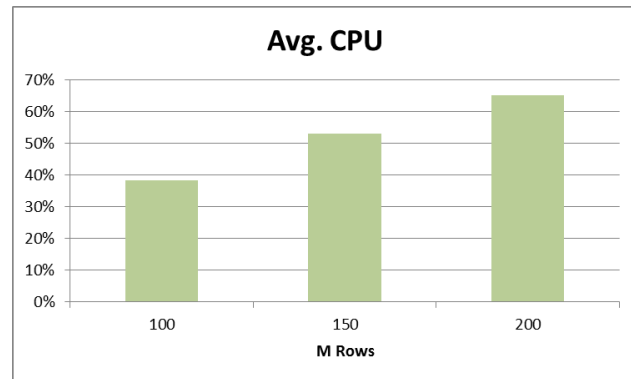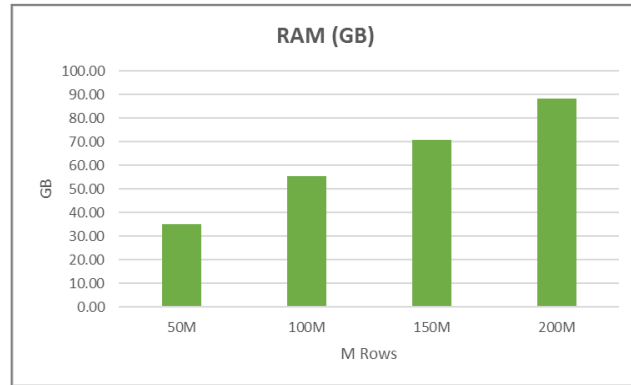
Enterprise Data layer

Qlik Sense enables you to leverage an optional data layer. These highly compressed files are called .QVD files. A benefit of having this intermediate data layer is reusability and incremental loads of large data sets. Several Qlik Sense applications can use the same data in the .QVD data layer without having to individually query remote data storages, and this process, along with Qlik Sense's full-featured ETL capability, are all automated by the Qlik Sense Scheduler component.

Source Data load performance

Qlik Sense can connect to almost any source to populate an application with data. The reload duration depends on several variables including CPU clock frequency, the data source, network speed, disk speed, types of aggregations, and calculations within load script. However, Qlik Sense scaling characteristics for reloading data shows a linear trend along several dimensions as the amount of data grows. This yields predictability for reload duration. The figures below show an example of measurements where different amounts of data have been loaded from a database. Both duration and RAM utilization scale linearly.

VISUALIZATION APPS

OPERATIONAL DATA SOURCES

Reload duration SQL

Peak RAM, SQL
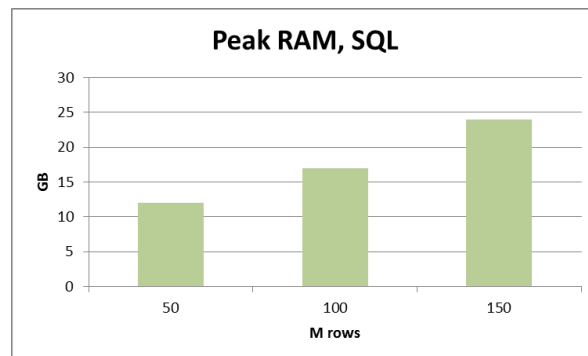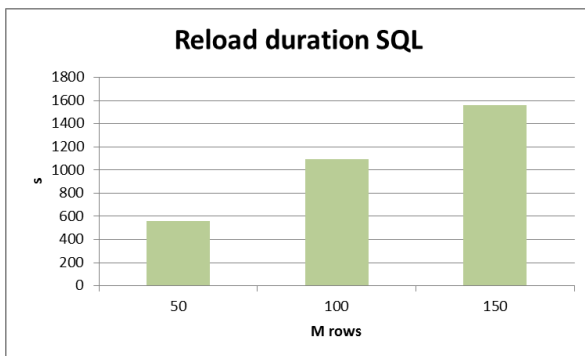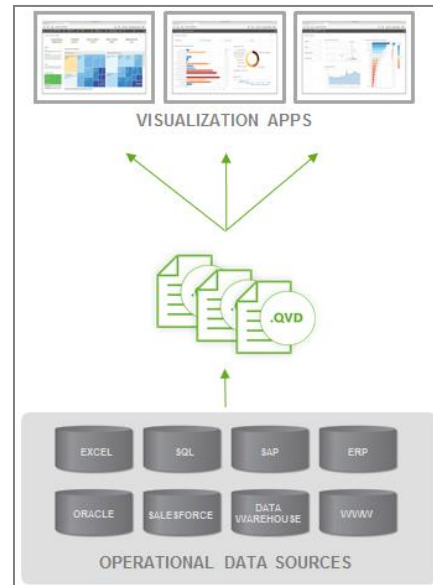
Data layer performance

The amount of time to load data from the Qlik Sense .QVD data layer is significantly shorter than querying the source. Loading data from the .QVD data layer also scales linearly with duration and RAM utilization.

Reload duration QVD


Peak RAM, QVD
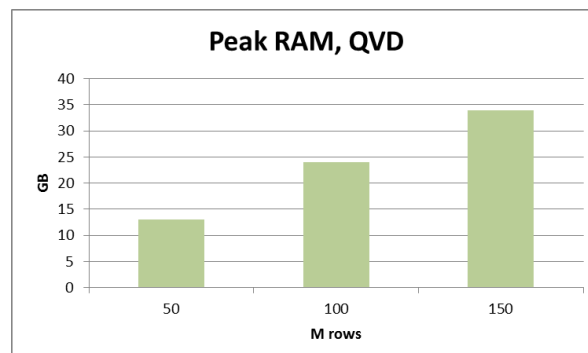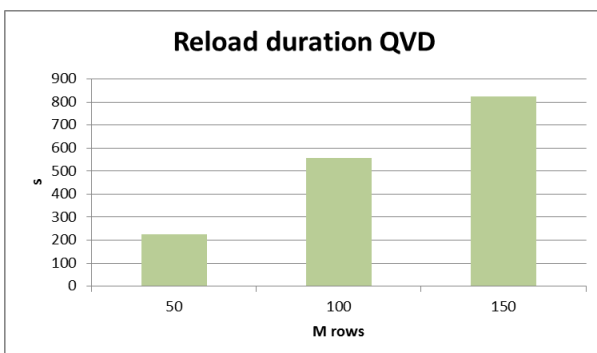
Scalable with users

Qlik Sense's performance scales uniformly with users. As new users are added to Qlik Sense, the performance impact is predictable and can be addressed by a proportional addition of more CPU and RAM resources. The following graphs show that as the number of concurrent users increase, the amount of CPU and RAM consumptions increases linearly, as well.

CPU

The Qlik Sense analytics engine is a 64-bit, multi-threaded process that is optimized to take advantage of multiple processor cores when performing calculations. When a user makes a selection, the Qlik Sense engine makes a short burst of intense CPU usage over all CPU cores in an effort to return results to a user as quickly as possible, and when not busy CPU utilization will approach 0%. The appropriate measure of CPU utilization and capacity, then, is the average CPU, and because of how the engine is designed, Qlik Sense can scale CPU utilization linearly as users are added to the platform.

RAM

The chart on the right illustrates principles highlighted in the section titled "Memory and Cache Management". As the number of concurrent users double, the RAM footprint increases linearly, but the actual amount of RAM used does not double. Application data is loaded a single time and is not duplicated for multiple users accessing and analyzing it. Additionally, note that RAM utilization actually slows as the number of users increase. This is an effect of increasing hits on the global cache.

Throughput

Avg. CPU

RAM (GB)
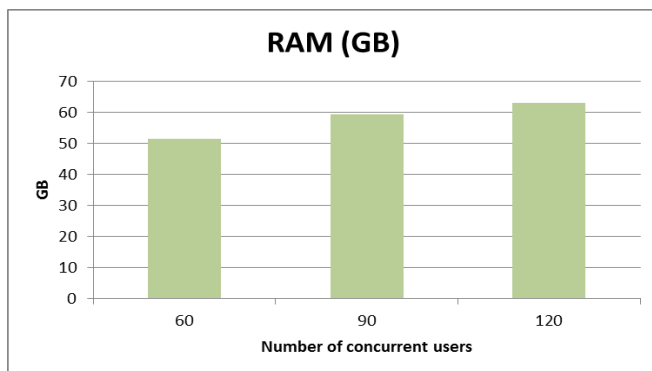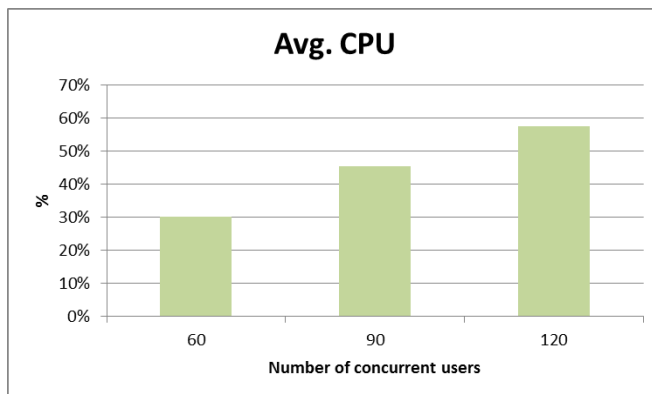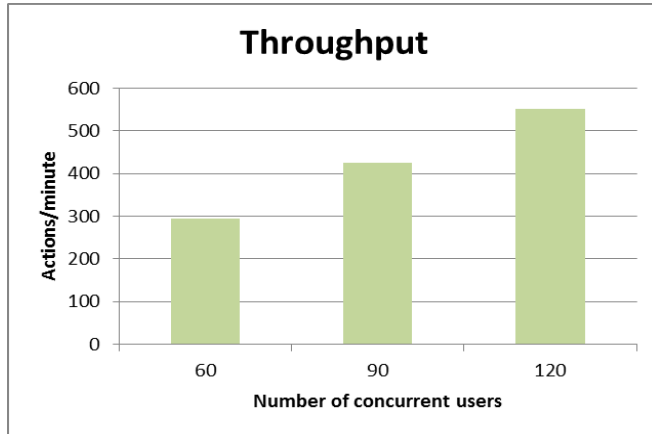
Scalable delivery

Qlik Sense enables your organization to align the work with the people most suited to do it. IT professionals can efficiently manage hardware by deploying Qlik Sense in virtual, cloud, physical environments, and this environment can be reconfigured, scaled up, out or scaled down as needed. IT professionals can control access to data sources through centrally managed data connections, and administrative capability may be delegated to line of business administrators to whatever level is appropriate for your organization.

Within the user community, content creators can combine and prepare data, build and use libraries of dimension and metrics, drag and drop visualizations, create stories, and publish content.

Qlik Sense enables your organization to deploy self-service at scale.

Summary

Qlik Sense provides consistent and predicable performance. Qlik Sense's architecture allows for horizontal and vertical scaling while scaling linearly with data, applications, and users. Qlik Sense allows IT professionals to effectively maintain end-user performance and capacity plan for future expected system usage.

To learn more, visit qlik.com.