

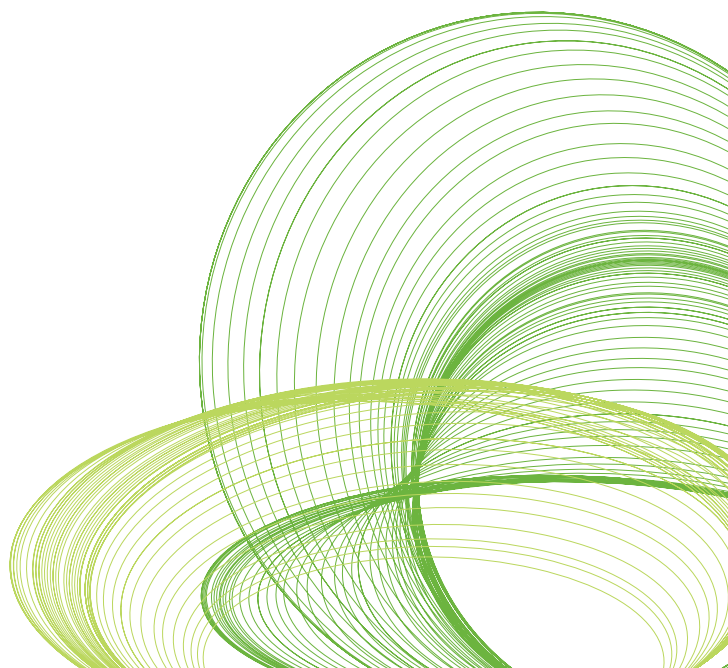
QlikView

QlikView 環境における スケールアップとスケールアウト

QlikView テクニカルブリーフ

2012年2月

qlikview.com



はじめに

企業の「Business Discovery」—膨大な情報の中からビジネスにおける課題と解決法を探索すること—の環境では、大量のデータやアプリケーションとともに、社内外のあらゆる数やタイプのユーザーにも対応できるよう、基礎となるアーキテクチャを効果的に拡張できることがますます重要になっています。

QlikTech スケーラビリティセンターは、パフォーマンスと拡張性の課題に特化して取り組んでいます。スケーラビリティセンターは、QlikViewのパフォーマンスに関連した問題を調査するツールやガイドラインを現場の方々に提供することを目的としています。また、QlikViewのパフォーマンスと拡張性についてさまざまなテストを実施し、このテーマに関する手引きを提供しています。本資料は、スケーラビリティセンターのテクニカルブリーフ・シリーズの一つです。

QlikViewを初めて導入する場合、または既存のQlikViewを拡張する場合に直面する課題の一つは、スケールアップ型アーキテクチャ（1台の大規模サーバー）とスケールアウト型アーキテクチャ（クラスタサーバー）のどちらを採用するかということです。このテクニカルブリーフは、クラスタサーバー環境とシングルサーバー環境におけるQlikView Serverのパフォーマンスを比較するために、QlikTech スケーラビリティセンターが実施した拡張性テストの概要を述べています。本資料の趣旨は、スケールアップかスケールアウトかを決定する際の考え方について、いくつかの例と一般的な見解を提供することであり、それらの結果は手引きとしてご利用いただくことを目的としています。QlikViewアプリケーションおよびデータタイプは複雑であることから、異なるパフォーマンス結果が出ることもあります。

このテクニカルブリーフではさらに、テストの方法論と構成について述べています。また、明らかになったパフォーマンスを説明するとともに、結果を要約しています。

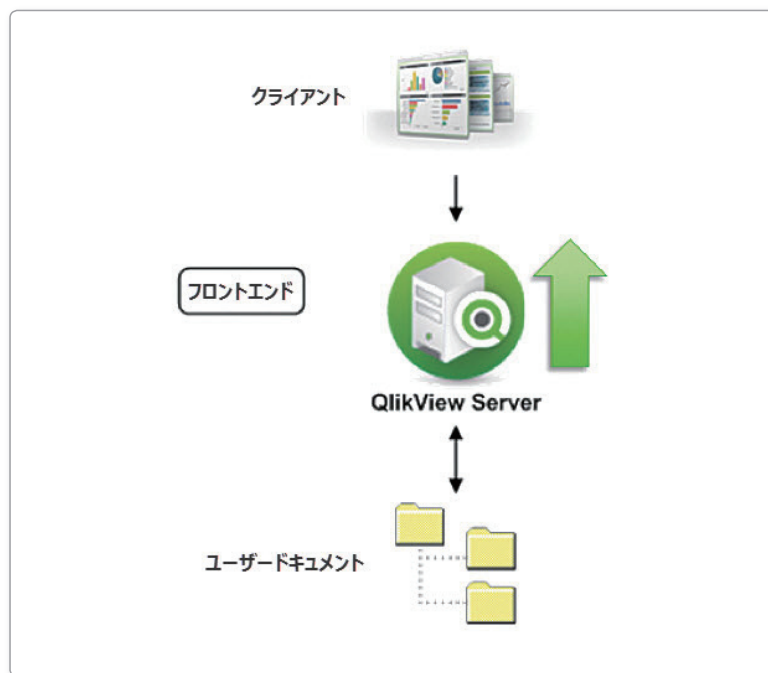
このテクニカルブリーフでは、障害リスク、フェイルオーバー計画、管理、ハードウェアコストなど、アーキテクチャの決定に影響を与える他の要素については論じていません。ハードウェア・アーキテクチャの選択は複雑であり、考慮すべき事柄が数多くあります。本資料は、特定のアーキテクチャの推奨または宣伝を意図したものではありません。

スケールアップ型アーキテクチャ (シングルサーバー) とは？

スケールアップ型アーキテクチャでは、シングルサーバーを使用して QlikView アプリケーションを実行します。

この場合、より多くのスループットが要求されるため、より大規模または高速なハードウェア (より大容量の RAM または CPU) が、同じサーバーに追加されます。

図表 1：スケールアップ型アーキテクチャ



スケールアウト型アーキテクチャ (クラスタサーバー) とは？

スケールアウト型アーキテクチャでは、パフォーマンス要件を満たすためにより多くのスループットが必要とされる場合、さらなるサーバーが追加されます。この種類のアーキテクチャでは、コモディティ・サーバー (汎用部品を採用して価格を抑えたミドルレンジサーバー) を使用するのが一般的です。スループットの必要性に伴い新たなサーバーが追加され、クラスタ型の QlikView 環境が形成されます。これらの環境では、QlikView Server が、複数の物理コンピュータまたは論理コンピュータにまたがる QlikView アプリケーションの負荷分散を行います。QlikView ロードバランシングは、事前定義されたアルゴリズムに従って、クラスタ内の負荷 (エンドユーザーのセッション) を分散する機能です。このアルゴリズムでは、どのノードがどのセッションを担当するかが定義されています。QlikView Server バージョン 11 は、3 種類のロードバランシングのアルゴリズムをサポートしています。

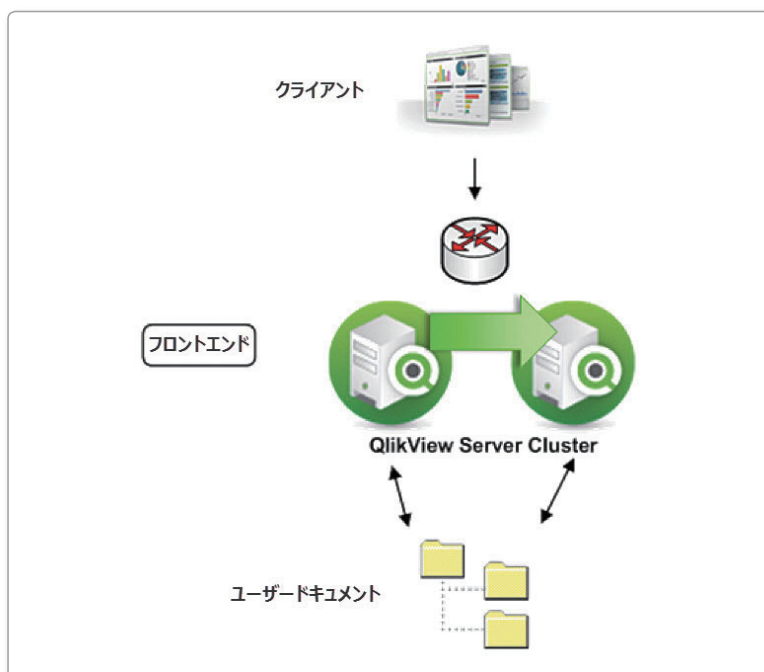
以下にそれぞれのスキームを簡単にまとめました。詳細については、ホワイトペーパー「QlikView スケーラビリティ概要 (QlikView Scalability Overview Technology)」をご参照ください。

- **Random** : デフォルトのロードバランシング・スキーム。ユーザーは、起動したい QlikView アプリケーションが QlikView Server にロードされているかどうかにかかわらず、ランダムにサーバーに割り当てられます。

- **Loaded document** : 特定のQlikViewアプリケーションが1台のQlikView Serverにだけロードされている場合、ユーザーはそのQlikView Serverに割り当てられます。アプリケーションが2台以上のQlikView Serverにロードされている、もしくはどのQlikView Serverにもロードされていない場合は、ユーザーはRAMの空き容量が最大のQlikView Serverに割り当てられます。
- **Cpu with Ram overload** : ユーザーは最もビジーでないQlikView Serverに割り当てられます。

このテクニカルブリーフでは、各ロードバランシング・アルゴリズムをいつ使用し、どのように調整して最高のパフォーマンスを得るかについての詳細には触れていないことをご了承ください。本資料で取り上げたクラスタテストは、特定のテストのある条件下で高いパフォーマンスが得られるよう構成された環境のもとで実施されました。

図表 2 : スケールアウト型アーキテクチャ



スケールアウトにするかスケールアップにするかを選択する際の考慮事項

通常QlikViewは、コアに対して非常に優れた拡張性があり、メモリも効率的に使用します。QlikViewは、キャッシュされた結果セット、アプリケーションそのもの、セッション状態を保存するために、利用可能なメモリを割り当てます。QlikView Serverは、QlikView Server (QVS) プロセスが、物理的にインストールされたRAMの一定量を使用できるように構成されています。

RAMが許容量を超えると、新たなQlikViewアプリケーションやセッション状態の情報が収まるよう、QVSプロセスはキャッシュされた結果セットの消去を開始します。

QlikViewはキャッシュされた結果セットに対し、許容メモリをすべて最速で割り振ることに留意してください。CPUの使用率と処理能力についても同様です。

CPU使用率がピーク中に高いのはよいことです。これは、そのアプリケーションがコアに対して優れた拡張性を持つよう設計されていることを意味します。選択と演算には、一定の処理能力（チップからのクロック周期）が必要であると考えられます。使用可能なコアすべてが協働して演算を行うために、高使用率のピーク時にはレスポンス時間が短縮されます。

スケールアップかスケールアウトを行って、環境の処理能力またはRAM容量を増加させることができます。この2つの解決策のどちらがより適切かは状況により異なります。アップグレードを計画する際に考慮すべきポイントを以下にまとめています。また、次のセクションでは、異なる条件下での実測値の例を示しています。

- クラスタ環境ではメモリを共有しません（各ノードは独自のRAM割り当てが必要です）。
 - ごく少数のユーザーが単一のアプリケーションを稼働している状態で起きるRAM不足の問題に関しては、スケールアウトで解決できる可能性は低いと考えられます。
 - 異なるノードに負荷分散が可能な複数のアプリケーションを稼働している環境では、たいていスケールアウトが最適な選択肢です。
- 多数のコアを搭載したサーバーは、コアあたりのクロック周波数が低いのが一般的ですが、こうしたチップセットには通常、物理的により多くのRAMをインストールできます。
 - より多くのコアを搭載したチップセットに対するスケールアップでは、必ずしも処理能力が向上するわけではありません。チップセットはそれぞれ異なる特色を持っています。同一ファミリーのチップセットの場合は、クロック周波数も考慮する必要があります。
 - 多数の同時ユーザーを考慮しなければならない場合は、コアの増設による処理能力の向上が効果的です。
- 多数の同時ユーザーに焦点を当てた単一アプリケーションをホストする環境では、単一サーバーに搭載された大量のRAMがキャッシュヒット率を高められるため、効果的である場合があります。
- 演算の観点から重いと考えられているアプリケーションが含まれるアプリケーション群については、クラスタが効果的です。これは、他のアプリケーションのパフォーマンスに悪影響を与えないよう、そのアプリケーションを別の専用ノードにロードできるためです。

スケールアップとスケールアウトのパフォーマンステスト比較

以下のセクションでは、2種類の状況下で行ったパフォーマンステストの結果についてまとめます。このテストの目的は、QlikView Serverのパフォーマンスをクラスタサーバーとシングルサーバーで比較し、その違いを示すことです。テストではQlikViewバージョン11を使用しました。

テストの方法論

この拡張性テストでは、負荷およびパフォーマンスのテストツールであるJMeterを使用して、異なるQlikViewアプリケーションにおけるユーザー・インタラクションのシナリオを作成しました。各シナリオでは、異なる環境（中規模のシングルサーバー、大規模のシングルサーバー、クラスタサーバー）において、同一の使用パターンをシミュレーションしています。テストシナリオの詳細は、各セクションに明記されています。

ハードウェアについて

2台のマシンで構成され、以下を搭載したQlikView Server クラスタに対してクラスタテストを実施しました。

- 12コア
- 3.33 GHz
- 144 GB RAM

アクセスポイントは、別のサーバー上のインターネット・インフォメーション・サービス (IIS) で稼働しています。中規模のシングルマシンのテストは、上記の仕様に従い、シングルマシンに対して実施されました。

大規模のシングルサーバーのテストは、以下の仕様に基づくサーバーに対して実施されました。

- 32コア
- 2.27 GHz
- 256 GB RAM
- IISおよびQlikView Serverプロセスは同一サーバー上で稼働。

2つの環境における処理能力を見積もるために、1秒間に利用可能なクロックサイクルを割り出しました。このような比較は、同一ファミリーのCPUに対してのみ有効です。この計測では、チップは同一のIntelファミリーに属しているため、2つの環境の処理能力が同程度であることが示されました。

表1：各環境における1秒あたりのクロックサイクル数

	クラスタサーバー	シングルサーバー
クロックサイクル/秒	80 G サイクル	73 G サイクル

シナリオ 1. 同時接続数が多い場合と少ない場合における大規模なQlikViewアプリケーションのパフォーマンステスト

このシナリオでは、同時接続数が多い場合と少ない場合という2つの異なる設定においてパフォーマンスを比較するために、2時間かけてテストを行いました。2つの類似したシナリオのもと、ユーザー・インタラクションをシミュレーションしています。この調査では、開発のベストプラクティスに基づいて設計された大規模なQlikViewアプリケーションを使用しました。QlikViewアプリケーションおよびテストシナリオの特徴は以下のとおりです。

表 2 : テストに使用した QlikView アプリケーションおよびシナリオの特徴

	アプリケーションのサイズ	QlikView アプリケーション内のレコード数	同時接続ユーザー数 (シミュレーション)	選択操作間の平均思考時間
シナリオ1 (同時接続数=少)	大	233 M	1ユーザー	15秒
シナリオ2 (同時接続数=多)	大	233 M	30ユーザー	15秒

テスト結果の概要は以下のとおりです。

表3：テスト結果

QlikView環境	シナリオ	平均レスポンス時間 (ms)/アクション	スループット [アクション(クリック)/分]
中規模のシングルサーバー	同時接続数=少	2042	2
	同時接続数=多	5446	61
クラスタサーバー (ロードバランス設定:CPU with RAM overload)	同時接続数=少	2231	2
	同時接続数=多	3034	65
大規模のシングルサーバー	同時接続数=少	1753	3
	同時接続数=多	1500	70

同時接続数が少ない場合のテスト結果

同時接続数が少ない場合のテストでは、QlikViewアプリケーションとのインタラクションに単一ユーザーを使用しました。結果では、処理能力とRAM容量の大きい大規模のシングルサーバーが、中規模のシングルサーバーに比べて著しく高速なレスポンスを実現していることが明確に示されています。大規模サーバーのパフォーマンスが高かった理由は、処理能力の増強、およびコアに対する拡張性が高い優れたアプリケーション設計によるものと判断できます。

またテスト結果では、この設定においてクラスタサーバーのパフォーマンスが低かったことも示されました。スケールアウトにより処理能力を増強した場合、ロードバランサーの中間層が原因となりオーバーヘッドが生じるためであり、単一ユーザーのシナリオでは自然な結果です。

同時接続数が多い場合のテスト結果

同時接続数が多いシナリオでは、大規模なシングルサーバーがより高いパフォーマンスを発揮しました。同時接続数が増えると、キャッシュヒット率が増加するためです。他のシナリオと比較すると、テストを実施した条件においては、スケールアップ・ソリューションが大きなパフォーマンス上の利点をもたらすことが明らかです。

同時接続数が多いシナリオでは、少ない場合のシナリオに比べ、クラスタサーバーがより高いパフォーマンスを発揮しました。しかしながら、クラスタソリューションはキャッシュ結果からはあまり恩恵を受けておらず、テストを実施した条件においては、スケールアップ・ソリューションほど高いパフォーマンスが得られていません。

同時接続数が多いシナリオにおけるCPU使用率の平均を以下の表にまとめています。
中規模のシングルサーバーは負荷テスト中に飽和状態となり、処理能力不足がレスポンス時間の遅れを引き起こしていることが分かります。

表4：高負荷シナリオにおけるCPU使用率の平均

QlikView環境	CPU QVS		平均CPU使用率
	ノード1	ノード2	
中規模の シングルサーバー	---	---	73%
クラスタサーバー	41%	44%	43%
大規模の シングルサーバー	---	---	41%

シナリオ2：異なる特徴を持つ3つのQlikViewアプリケーション

このテストでは、3つの異なるQlikViewアプリケーションに対するユーザー・インタラクションをシミュレーションした3つのシナリオを使用しました。異なるアプリケーションに対するテストを10分間隔で開始し、望ましい負荷で1時間同時に実行しました。3つのQlikViewアプリケーションおよびテストシナリオの特徴は以下のとおりです。

表5：テストシナリオの詳細

	アプリケーションの サイズ	QlikView アプリケーション内の レコード数	同時接続ユーザー数 (シミュレーション)
シナリオ1	小	80.000	40 ユーザー
シナリオ2	小	100	50 ユーザー
シナリオ3	大	600.000.000	20 ユーザー

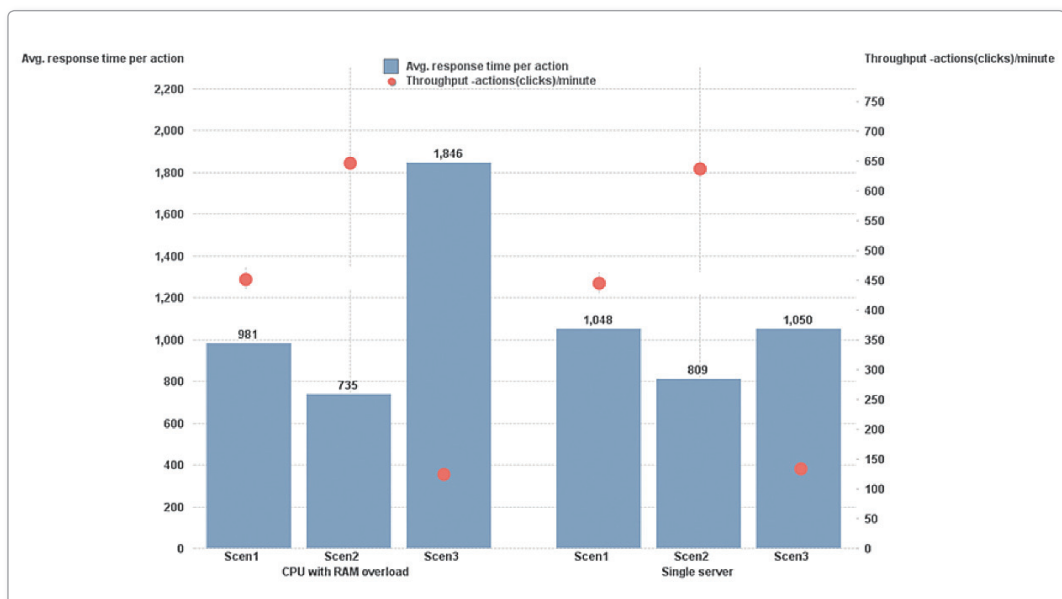
QlikView Serverは各テストを実行する前に再起動されているため、QlikViewアプリケーションはメモリにプリロードされていません。

以下の図表は、クラスター環境およびシングルサーバー環境で各アプリケーションを開いた結果を示しています。クラスター環境では、CPU with RAM overloadのクラスタリング・アルゴリズムを使用しました。以下に示す結果は、すべてのアプリケーションが並行にロードされたテスト時間中から抽出したものです。

表6：クラスターサーバー環境とシングルサーバー環境のQVSパフォーマンスを比較した、平均レスポンス時間およびスループット結果

QlikView環境	シナリオ	平均レスポンス時間(ms)/ アクション	スループット [アクション(クリック)/分]
クラスターサーバー (ロードバランス設定: CPU with RAM overload)	シナリオ1	981	451
	シナリオ2	735	647
	シナリオ3	1846	122
大規模のシングルサーバー	シナリオ1	1048	445
	シナリオ2	809	637
	シナリオ3	1050	133

表7：「CPU with RAM overload」設定のクラスターサーバー環境、およびシングルサーバー環境を比較したパフォーマンステスト結果



テスト結果から、小規模なQlikViewアプリケーションでは、シングルマシン環境に比べてクラスター化されたQlikView環境の平均レスポンス時間が短いことがわかります。大規模なQlikViewアプリケーションでは、平均レスポンス時間が短かったのはシングルマシン環境でした。クラスター環境およびシングルマシン環境におけるスループットの差異は、いくつかのリクエストによるものです。全体的なスループット値は以下のとおりです。

表8 : 「CPU with RAM overload」設定のクラスタサーバー環境、およびシングルサーバー環境を比較したスループット結果

	スループット [アクション(クリック)/分]
クラスタサーバー (ロードバランス設定: CPU with RAM overload)	1221
シングルサーバー	1215

テスト中のCPU使用率の比較は以下のとおりです。

表9 : クラスタ環境とシングルサーバー環境を比較したCPU使用率結果

QlikView環境	CPU QVS		平均CPU使用率
	ノード1	ノード2	
クラスタサーバー (ロードバランス設定: CPU with RAM overload)	46%	52%	49%
シングルサーバー	---	---	47%

調査したハードウェアの設定では、「CPU with RAM overload」ロードバランシング・アルゴリズムを用いたクラスタサーバー環境、およびシングルサーバー環境は、全体的なスループットが似通ったものでした。主な違いは、大規模なQlikViewアプリケーションに対して、大規模マシンのシングルサーバー環境が高いパフォーマンスを発揮したことです。その理由は、キャッシュのためにより多くのRAMを使用できるためであるといえます。クラスタ環境は、同一のQlikViewアプリケーションに対して低いパフォーマンス結果を出しています。これは、クラスタサーバーはノードあたりの使用可能なRAMが少ないためです。また、この大規模なQlikViewアプリケーションはコアに対する拡張性に優れているため、小規模なクラスタマシンでは演算中に処理能力が飽和してしまう量の処理をこなす必要があることも、この結果を説明しています。

まとめ

スケールアップ型またはスケールアウト型アーキテクチャのどちらに利点があるかは、状況により異なることが、今回実施した2例のテストの結果から明らかになりました。

これらのテストは、大量のデータセットとともに、適切かつ大規模な演算がQlikViewアプリケーションに含まれる場合、通常は大規模なサーバーがより優れたパフォーマンスを発揮することを示しています。

また、大規模なマシンは一般的に、RAMのスケールアップによく対応しています（小規模なマシンにはハードウェアの制約があります）。その反面、小規模なマシンはたいていCPUのクロック周波数が高くなっています。同時接続ユーザーからのリクエストが多い場合は、クラスタ環境の小規模サーバーが、大規模サーバーと同様、あるいはそれ以上のパフォーマンスを発揮することも多くあります。

References

QlikView Development and Deployment Architecture Technical Brief

www.qlikview.com/.../global-us/direct/datasheets/DS-Technical-Brief-Dev-and-Deploy-EN.ashx

QlikView Architecture and System Resource Usage Technical Brief

www.qlikview.com/.../DS-Technical-Brief-QlikView-Architecture-and-System-Resource-Usage-EN.ashx

QlikView Scalability Overview Technology White Paper

<http://www.qlikview.com/us/explore/resources/whitepapers/qlikview-scalability-overview>

© 2012 QlikTech International AB. All rights reserved. QlikTech, QlikView, Qlik, Q, Simplifying Analysis for Everyone, Power of Simplicity, New Rules, The Uncontrollable Smile and other QlikTech products and services as well as their respective logos are trademarks or registered trademarks of QlikTech International AB. All other company names, products and services used herein are trademarks or registered trademarks of their respective owners. The information published herein is subject to change without notice. This publication is for informational purposes only, without representation or warranty of any kind, and QlikTech shall not be liable for errors or omissions with respect to this publication. The only warranties for QlikTech products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting any additional warranty.