

# 1 Möglichkeiten zur Verbesserung der Performance in Stammdialogen/Listenansichten des Auftrag SQL

Beim Einsatz von Auftrag SQL treten immer wieder Fragen zur Geschwindigkeit der Listenansichten und zu Verbesserungsmöglichkeiten auf. In diesem Dokument wird zum einen der technische Hintergrund der SQL Server-Abfragen erklärt und zum anderen werden die im Auftrag vorhandenen Möglichkeiten zur Beschleunigung der Datenanzeige vorgestellt.

## 1.1 Technischer Hintergrund

Die Vorgehensweise zum Anzeigen einer grösseren Datenmenge in der SQL-Version unterscheidet sich grundlegend von der in der BDE-Version eingesetzten. Die Unterschiede resultieren aus der Arbeitsweise der Datenbanken: Die BDE erlaubt einen dateibasierten Zugriff auf die benötigten Daten, der SQL Server liefert die Daten basierend auf einer entsprechenden Abfrage. In der BDE-Version ist es deshalb möglich, nur die für die Anzeige benötigten Daten aus der Tabellendatei auszulesen und erst beim Scrollen/ Weiterblättern in den Listen die nächsten Datensätze zu lesen und anzuzeigen. Dadurch stehen die ersten Sätze selbst einer umfangreichen Liste sofort zur Verfügung.

Die SQL-Version bekommt ihre Daten aufgrund einer Abfrage beim SQL Server, d.h. für eine ungefilterte Listenansicht sind alle Datensätze der zugrunde liegenden Tabelle beim Server abzufragen. Im Ergebnis wird die gesamte Datenmenge auf dem SQL Server zusammengestellt und an den Auftrag übertragen. Während also die BDE-Version die ersten 20 oder 30 Zeilen der Liste abfragt und anzeigt, liefert der SQL Server u. U. Tausende von Zeilen zurück. Um dies zu verhindern wurde das „Paging“ für die SQL-Version entwickelt. Es sorgt dafür, dass nur die für das „Paging“ angegebene Anzahl von Datensätzen vom Server abgefragt und zur Anzeige gebracht wird. Wird der abgefragte Datenbereich verlassen (z. B. durch Blättern oder Scrollen), werden jeweils die nächsten Daten vom Server abgefragt.

## 1.2 Programminterne Möglichkeiten zur Performanceverbesserung

Es gibt allgemeine „Stellschrauben“ und eine spezielle für die Artikelliste:

1. Die Artikelliste zeigt abhängig von den Mandanteneinstellungen bis zu 3 Preise und den Bestand der Artikel mit an. Die Preise und Bestände werden jedes Mal beim Abfragen der Artikeldaten vom SQL Server neu berechnet. Die Preisberechnung erfordert abhängig von der Anzahl und Struktur der abgefragten Artikel (Anzahl von Stücklisten, hohe Verschachtelungstiefe der Stücklisten) sehr viel Zeit. Deshalb sollten in den Mandanteneinstellungen unter „Artikelauswahl“ wirklich nur die Preise eingestellt werden, die in der Listenansicht unbedingt benötigt werden.
2. In den Einstellungen der Stammdialoge befindet sich darüber hinaus unter „Schlüssel“ der Abschnitt „Bei ungültigem Datensatz...“, in dem festgelegt werden kann, was nach der Eingabe eines ungültigen Schlüssels in ein Auswahlfeld passieren soll: Die Standardeinstellung „...eine Auswahlliste zeigen“ öffnet die Listenansicht und lässt den Benutzer einen Datensatz aus der Liste auswählen. Interessant im Hinblick auf schnelles Arbeiten sind die Einstellungen „...filtern in allen Feldern“ und „...filtern in einem Feld“. Das Filtern in allen Feldern trägt den vom Benutzer bereits eingegebenen (Teil-)Schlüssel automatisch in den Quickfilter der Listenansicht ein. Die Abfrage würde also die Benutzereingabe verwenden, um nur zu dieser Eingabe passende Datensätze anzuzeigen. Eine strengere und damit auch schnellere Filterung erfolgt mit der Einstellung „...filtern in einem Feld“: hier kann ein einzelnes Feld vorgegeben werden, in dem die Benutzereingabe gesucht wird. Beispiel.: Der Benutzer erfasst Positionen in Belegen. Bei fehlerhafter oder unvollständiger Eingabe einer

Artikelnummer öffnet sich standardmässig die Artikelliste mit allen vorhandenen Artikeln. Mit der Einstellung „...filtern in einem Feld“ würde die Eingabe des Benutzers im Artikelnummernfeld des Belegdialogs automatisch in den Quickfilter der Artikelliste übertragen werden und die Abfrage anhand des Quickfilters für das angegebene Feld ausgeführt werden.

3. Eine entscheidende Bedeutung für die Geschwindigkeit besitzt das Sortierkriterium der jeweiligen Liste, in der ausgewählt werden soll. Wird beispielsweise die Artikelliste der Belegpositionserfassung nach dem Feld Bezeichnung aufsteigend sortiert aufgerufen, erhöht der nachfolgende Index die Geschwindigkeit:  
`create index ARbezAb on art (bezeichnung desc)`  
Wird nach demselben Feld allerdings absteigend sortiert, ist ein weiterer Index erforderlich:  
`create index ARbezAb on art (bezeichnung desc)`  
Das bedeutet, es sollten gemäss den Anforderungen des Anwenders nachträglich Indizierungen für die wichtigsten Tabellen im SQL Server eingerichtet werden. Es sollte allerdings in keinem Fall daraus geschlossen werden, dass es sinnvoll ist, im SQL Server willkürlich auf alle Felder einer Tabelle Indizes zu erzeugen. Die maximale Anzahl der möglichen Indizes, Felder und Statistiken sind im SQL Server pro Tabelle begrenzt. Deshalb ist anzuraten, nur wirklich notwendige Indizes zu erzeugen. Weiterhin ist zu beachten, dass die Indizes verlorengehen, wenn die Tabelle innerhalb eines Updates reorganisiert werden muss (betrifft im Normalfall nur Vollversionen, in sehr seltenen Ausnahmefällen Servicepacks).
4. Ein weiterer Performancevorteil wird erlangt, wenn auf die Anzeige der Datensatzanzahl in der Statuszeile des Dialogs verzichtet werden kann. Die Anzeige kann mit nachfolgendem Datenbankeintrag verhindert werden:  
`insert into [SL_MUFAKT].[dbo].inifiles (iniFileName,block,eintrag,wert)  
values ('MANDANT.INI','EINSTELLUNGEN','DisableCountPanel','1')`

## 2 Zusammenfassung

Es gibt verschiedene Ansatzpunkte zum schnelleren Aufbau der Listenansichten. Besonderes Augenmerk sollte auf die Artikelliste gelegt werden: sie wird sehr häufig benutzt und benötigt aber u. U. relativ viel Zeit zur Berechnung der angeforderten Daten. Hier ist im Gespräch mit dem Kunden herauszuarbeiten, ob die Anzeige der Preise eingeschränkt werden kann. Das grösste Potential zur Verbesserung der Performance hat aber der Anwender selbst: er hat die Möglichkeit, die Anzahl der vom Server abgefragten Daten durch die „Paging“- Einstellungen einzuschränken. Die Unterstützung des Benutzers beim Filtern der Datensätze durch die Listeneinstellungen (bei ungültigem Schlüssel filtern in...) sollte unbedingt in Betracht gezogen werden.

17.03.2014/ld/V1.0