

**Unica Interact Version 12.1.1  
Optimierungshandbuch**



# Inhalt

<b>Kapitel 1. Optimieren von Unica Interact für optimale Leistung.....</b>	<b>1</b>
Cacheverwaltung.....	1
Arbeiten mit Ehcache.....	2
Mit ignite Arbeiten.....	7
Unica Interact-API.....	10
Installation und Netzkonfiguration.....	10
Verwalten interaktiver Ablaufdiagramme.....	10
Serviceoptimierung.....	12
Optimieren der Webanwendungsserver.....	12
JVM-Argumente.....	12
Verbindungspool.....	14
Terminologie der Optimierung.....	15
Datenbankoptimierung.....	16
Optimieren des ETL-Prozesses.....	16
Protokollierung.....	21

# Kapitel 1. Optimieren von Unica Interact für optimale Leistung

Eine Installation von Unica Interact besteht aus mehreren Komponenten, einschließlich Tools von anderen Anbietern (wie Webanwendungsserver, Datenbanken und Lastausgleichsmodule) und -Komponenten wie Unica Platform und Unica Campaign. Die verschiedenen Eigenschaften, Funktionen und Einstellungen dieser Komponenten können Sie zur Optimierung der Leistung konfigurieren.

Mit den Konfigurationseigenschaften von Unica Interact selbst können Sie Ihre Installation so anpassen, dass die optimale Leistung erzielt wird.

Die Definition von "optimale Leistung" ist jedoch schwierig. Jede Umgebung, jede Implementierung stellt andere Anforderungen. Eine Implementierung von Unica Interact, bei der z. B. die Daten für interaktive Ablaufdiagramme anhand von Echtzeitdaten zusammengestellt werden, wird auf andere Weise optimiert, wie eine Implementierung, die Daten aus verschiedenen Datenbanktabellen benötigt.

Die Laufzeitleistung von Unica Interact wird von verschiedenen Faktoren beeinflusst, darunter die Hardwarekonfiguration, Netzkonfiguration und die Konfiguration von Unica Interact. Die folgenden Richtlinien und Empfehlungen können sich daher unterschiedlich auf Ihre Umgebung auswirken.

Die folgenden Richtlinien werden von den zugehörigen Komponenten verwaltet. Die Reihenfolge, in der Sie die Einstellungen ändern, ist unerheblich.

Bei Simulator

- Der Simulator sollte in einer Test-/Staging-Umgebung ausgeführt werden, da der Simulator mit einer höheren Anzahl von Threads läuft und auf dem Laufzeitserver viel CPU verbrauchen kann.

## Cacheverwaltung

In einer umfangreichen Unica Interact-Umgebung, in der Sie eine große Anzahl Laufzeitserver verwenden, können Sie Cacheverwaltungssoftware verwenden, um die

Laufzeitarbeitslast zu verteilen und die Echtzeitleistung der gesamten Laufzeitservergruppe zu verbessern. Wenn Sie die Unica Interact-Lösung aktivieren, können Sie mehrere Laufzeitserver verwenden, denen ein Programm für den Lastausgleich vorgeschaltet ist. Das Programm für den Lastausgleich gleicht die Arbeitslast der Laufzeitserver in der Gruppe aus und hilft, die Sitzungsaffinität zu verwalten. Das bedeutet, wenn eine eingehende Sitzung vom Laufzeitserver A bearbeitet wird, werden zusätzliche Anfragen desselben Benutzers von der Sitzung auf Server A erfüllt. Diese Methode der Balancierung der Verbindungen zwischen den Laufzeitservern in der Gruppe kann die Leistung erheblich verbessern, unterliegt aber auch einigen Einschränkungen, da die gesamten Sitzungsdaten im Speicher verwaltet werden und die Menge der verwalteten Sitzungsdaten durch die Speicherbegrenzung jeder Java™ Virtual Machine (JVM) einschränkt ist. Unica Interact unterstützt standardmäßig zwei Cache-Verwaltungslösungen:

- **Ignite:** Es ist eine speicherzentrierte, verteilte Cachelösung, die mit Interact geliefert wird. Weitere Einzelheiten zu Ignite finden Sie unter <https://apacheignite.readme.io/docs/what-is-ignite>
- **Ehcache** - Es handelt sich um eine Open-Source-Cachinglösung, die mit jeder Installation von Interact enthalten ist. Weitere Informationen zur Ehcache-Software finden Sie unter: <http://www.ehcache.org/documentation/>

## Arbeiten mit Ehcache

Um die Leistung Ihrer Unica Interact-Laufzeitservergruppe zu verbessern, können Sie Ehcache konfigurieren, indem Sie eine Reihe von Konfigurationseigenschaften in Unica Platform ändern.

Um Ehcache als Cache-Manager für jeden Unica Interact-Laufzeitserver zu aktivieren, müssen Sie die folgenden Konfigurationseigenschaften in Unica Platform auf `EHCACHE` festlegen:

```
Unica Interact Interact | cacheManagement | Cache-Manager | Ehcache
```

Um Ehcache als Cache-Manager für das Speichern von Ereignismusterzuständen für eine Laufzeitservergruppe zu verwenden, müssen Sie den folgenden Parameter ebenfalls auf `EHCACHE` festlegen:

Unica Interact > cacheManagement > caches > Interact cache > cacheManagerName

Sie müssen diesen Prozess auf jedem Unica Interact-Laufzeitserver in der Servergruppe durchführen, um Ehcache als Cache-Manager für die Servergruppe zu aktivieren.

Wenn Sie Ehcache als Cache-Manager aktiviert haben, können Sie die Einstellungen konfigurieren, um das Caching für Ihre Installation zu optimieren.

## Ändern der Konfigurationseinstellungen von Ehcache

Wenn Sie die Verwendung des integrierten Cache-Manager Ehcache zur Leistungsverbesserung für den Unica Interact-Laufzeitserver festlegen, können Sie die Einstellungen von Ehcache so konfigurieren, dass der Wert für die Laufzeitservergruppe optimiert wird.

Um den Cache-Manager Ehcache zu konfigurieren, können Sie die folgenden Konfigurationseigenschaften in Unica Platform öffnen:

Unica Interact Interact | Cache Managers | EHCACHE | Parameter Data

Diese Konfigurationskategorie enthält einen Satz Standardkonfigurationseigenschaften für Ehcache, die den Einstellungen entsprechen, die Sie in der Ehcache-Konfigurationsdatei angeben. Durch Klicken auf **(Parameter)** und Benennung entsprechend dem Ehcache-Parameter, den Sie ändern wollen, können Sie zusätzliche Parameter in dieser Kategorie erstellen.

Weitere Informationen zu den Konfigurationseigenschaften für Ehcache finden Sie im Unica Interact-Administratorhandbuch, Anhang B: Unica Interact Laufzeitumgebung - Konfigurationseigenschaften In der Dokumentation zu Ehcache (<http://www.ehcache.org/documentation/>) finden Sie ebenfalls weitere Informationen zum Thema.

Beachten Sie, dass Sie für die Leistungsoptimierung mit Ehcache in den Unica Platform Konfigurationseinstellungen für den Unica Interact-Laufzeitserver das Sitzungszeitlimit (`Interact > cacheManagement > caches InteractCache > TimeoutInSecs`) auf den kleinsten zulässigen Wert einstellen müssen.

Jede Unica Interact-Sitzung enthält eine bestimmte Menge an Sitzungsdaten im Speicher. Je länger Sie die Sitzungen aufrechterhalten, desto mehr paralleler Speicherbedarf ist erforderlich. Wenn Sie z. B. 50 Sitzungen pro Sekunde erwarten und jede Sitzung 20 Minuten

aktiv bleiben kann, muss der Speicher in der Lage sein, 60.000 Sitzungen gleichzeitig zu unterstützen, wenn jede Sitzung volle 20 Minuten dauert.

Der Wert muss für jedes Szenario logisch sein. Eine Aufruf-Systemsitzung sollte beispielsweise eine Minute aktiv sein, eine Website-Sitzung dagegen 10 Minuten.

## Unterstützen einer größeren Anzahl paralleler Sitzungen in Ehcache

In einigen Unica Interact-Umgebungen kann eine hohe Anzahl paralleler Sitzungen bei Verwendung von Ehcache als Cache-Manager dazu führen, dass die Unica Interact-Laufzeitumgebung den verfügbaren Hauptspeicher überschreitet und dadurch eine Systemverzögerung oder ein Fehler durch abnormale Speicherbedingungen verursacht wird. Abnormale Speicherbedingungen sind wahrscheinlich, wenn Sie den Konfigurationsparameter `maxEntriesInCache(Unica Interact > cacheManagement > Caches > InteractCache > maxEntriesInCache)` auf einen höheren Wert als die Standardeinstellung festgelegt haben. Allerdings kann eine solche Situation auch bei der Standardeinstellung von maximal 100.000 Sitzungen auftreten. Reduzieren Sie entweder den Wert für **maxEntriesInCache** oder ändern Sie anhand der folgenden Anweisungen den Cache des Systemspeichers, um die zwischengespeicherten Daten auf den Plattenspeicher zu übertragen. Mit dieser Änderung sind sehr viel mehr parallele Sitzungen möglich.

Um zu verhindern, dass die Unica Interact-Laufzeitumgebung den verfügbaren Hauptspeicher im Heapspeicher der Java™ Virtual Machine (JVM) überschreitet, können Sie den Caching-Mechanismus so ändern, dass zum Zwischenspeichern der Daten, die den verfügbaren Hauptspeicher überschreiten, der Plattenspeicher verwendet wird.

Systemadministratoren können die den Bereitstellungssystemen verfügbare Speicherkapazität über die folgenden JVM-Parameter anpassen:

```
-Xms####m -Xmx####m -XX:MaxPermSize=256m
```

Wobei die Zeichen #### 2048 oder höher sein sollten (abhängig von ihrer Systembelastung). Beachten Sie, dass ein 64-Bit-Anwendungsserver und eine JVM normalerweise für Werte größer als 2048 erforderlich sind.

Unica Interact verwendet zum Zwischenspeichern von Daten das verteilte Open-Source-Cache-System Ehcache. Unica Interact verwendet standardmäßig die von Unica Platform angegebenen Einstellungen, um das Caching mit Ehcache zu verwalten. Sie können diese Einstellungen für Unica Interact aber auch überschreiben und Ihre eigene Ehcache-Konfigurationsdatei erstellen, die beim Start von Unica Interact automatisch geladen wird.

Damit die benutzerdefinierte Ehcache-Konfigurationsdatei beim Start geladen wird, müssen die folgenden Bedingungen erfüllt sein:

- Die JVM muss die Eigenschaft `interact.ehcache.config` enthalten, wie im folgenden Beispiel dargestellt:

```
-Dinteract.ehcache.config=/temp/abc.xml
```

Sie können eine JVM-Eigenschaft für Ihren Webanwendungsserver entweder im Script des Startbefehls (Oracle WebLogic) oder in der Administrationskonsole einstellen (IBM® WebSphere®). Die Angaben in `/temp/abc.xml` geben den aktuellen Pfad zu der XML-Datei mit der Ehcache-Konfiguration an, die Sie beim Systemstart laden möchten.

- An der Position, die von der JVM-Eigenschaft angegeben wird, muss eine Konfigurationsdatei vorhanden sein, die die gültigen Konfigurationseinstellungen für Ehcache im XML-Format enthält.

Wenn Sie diese Eigenschaft nicht angeben oder wenn sich an der angegebenen Position keine Konfigurationsdatei befindet, verwendet Unica Interact die standardmäßige Cachekonfiguration.

Wenn beide Bedingungen zutreffen, wird die Ehcache-Konfigurationsdatei beim Start geladen. Die benutzerdefinierten Einstellungen überschreiben dann alle Standardkonfigurationsparameter von Unica Interact zum Caching der Sitzungsdaten.

Das folgende Beispiel ist eine Musterkonfigurationsdatei (im XML-Format), die Sie zum Anpassen von Ehcache verwenden können:

```
<ehcache xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation=" ../.. /main/config/ehcache.xsd">
```

```
<defaultCache
maxElementsInMemory="10"
eternal="false"
timeToIdleSeconds="5"
timeToLiveSeconds="10"
overflowToDisk="true"
/>

<cache name="InteractCache"
maxElementsInMemory="5"
eternal="false"
timeToIdleSeconds="600"
timeToLiveSeconds="600"
overflowToDisk="true">

<cacheEventListenerFactory
class="com.unicacorp.interact.cache.ehcache.EHCacheEventListenerFactory"
properties=""/>
<!--For InteractCache, use the following to use the
EHCacheEventListenerFactory:-->
<cacheEventListenerFactory
class="com.unicacorp.interact.cache.EHCacheEventListenerFactory"
properties=""/>
<!--For PatternStateCache, use the following to use
the EHCacheEventListenerFactory:-->
<cacheEventListenerFactory
class="com.unicacorp.interact.cache.PatternStateCacheEventListenerFactory"
properties=""/>
</cache>

</ehcache>
```



Wenn Sie diese Datei als `/HCL/Unica Interact/conf/Ehcache.xml` gespeichert haben, legen Sie die JVM-Eigenschaft für die Webanwendung entsprechend dem folgenden Beispiel fest:

```
-Dinteract.ehcache.config=/HCL/Unica Interact/conf/Ehcache.xml
```

Für eine vollständige Liste der verfügbaren Optionen zur Modifizierung der Ehcache-Software siehe <http://www.ehcache.org/documentation/>

## Mit ignite Arbeiten

Um die Leistung Ihrer Unica Interact-Laufzeitservergruppe zu verbessern, können Sie Ignite konfigurieren, indem Sie eine Reihe von Konfigurationseigenschaften in Unica Platform ändern.

Um Ignite als Cache-Manager für jeden Unica Interact-Laufzeitserver zu aktivieren, müssen Sie die folgenden Konfigurationseigenschaften in Unica Platform auf `Ignite` festlegen:

```
Unica Interact Interact | cacheManagement | Cache-Manager | Ehcache
```

Um Ignite als Cache-Manager für das Speichern von Ereignismusterzuständen für eine Laufzeitservergruppe zu verwenden, müssen Sie den folgenden Parameter ebenfalls auf `Ignite` festlegen:

```
Unica Interact > cacheManagement > caches > Interact cache > cacheManagerName
```

Sie müssen diesen Prozess auf jedem Unica Interact-Laufzeitserver in der Servergruppe durchführen, um Ignite als Cache-Manager für die Servergruppe zu aktivieren.

Wenn Sie Ignite als Cache-Manager aktiviert haben, können Sie die Einstellungen konfigurieren, um das Caching für Ihre Installation zu optimieren.

## Ändern der Ignite-Konfigurationseinstellungen

Wenn Sie die Verwendung des integrierten Cache-Manager Ignite zur Leistungsverbesserung für den Unica Interact-Laufzeitserver festlegen, können Sie die Einstellungen von Ignite so konfigurieren, dass der Wert für die Laufzeitservergruppe optimiert wird.

Um den Cache-Manager Ignite zu konfigurieren, können Sie die folgenden Konfigurationseigenschaften in Unica Platform öffnen:

`Unica Interact > cacheManagement > Cache Managers > EHCACHE > Parameter Data`

Diese Konfigurationskategorie enthält einen Satz Standardkonfigurationseigenschaften für Ignite, die den Einstellungen entsprechen, die Sie in der Ignite-Konfigurationsdatei angeben. Durch Klicken auf **(Parameter)** und Benennung entsprechend dem Ignite-Parameter, den Sie ändern wollen, können Sie zusätzliche Parameter in dieser Kategorie erstellen.

Weitere Informationen zu den Konfigurationseigenschaften für Ignite finden Sie im Unica Interact-Administratorhandbuch, Anhang B: Unica Interact Laufzeitumgebung - Konfigurationseigenschaften In der Dokumentation zu Ignite (<https://apacheignite.readme.io/docs>) finden Sie ebenfalls weitere Informationen zum Thema.

Beachten Sie, dass Sie für die Leistungsoptimierung mit Ignite in den Unica Platform Konfigurationseinstellungen für den Unica Interact-Laufzeitserver das Sitzungszeitlimit (`Interact > cacheManagement > caches InteractCache > TimeoutInSecs`) auf den kleinsten zulässigen Wert einstellen müssen.

Jede Unica Interact-Sitzung enthält eine bestimmte Menge an Sitzungsdaten im Speicher. Je länger Sie die Sitzungen aufrechterhalten, desto mehr paralleler Speicherbedarf ist erforderlich. Wenn Sie z. B. 50 Sitzungen pro Sekunde erwarten und jede Sitzung 20 Minuten aktiv bleiben kann, muss der Speicher in der Lage sein, 60.000 Sitzungen gleichzeitig zu unterstützen, wenn jede Sitzung volle 20 Minuten dauert.

Der Wert muss für jedes Szenario logisch sein. Eine Aufruf-Systemsitzung sollte beispielsweise eine Minute aktiv sein, eine Website-Sitzung dagegen 10 Minuten.

## Unterstützung einer größeren Anzahl gleichzeitiger Sitzungen in Ignite

Wenn Sie in einigen Unica Interact-Umgebungen Ignite als Cache-Manager verwenden, kann eine hohe Anzahl gleichzeitiger Sitzungen dazu führen, dass die Unica Interact-Laufzeit den verfügbaren Speicher überschreitet, wodurch es zu einer Verlangsamung des Systems oder zu einem Speicherausfall kommt. Abnormale Speicherbedingungen sind wahrscheinlich, wenn Sie den Konfigurationsparameter `maxEntriesInCache`(`Unica Interact >`

cacheManagement > Caches > InteractCache > maxEntriesInCache) auf einen höheren Wert als die Standardeinstellung festgelegt haben. Allerdings kann eine solche Situation auch bei der Standardeinstellung von maximal 100.000 Sitzungen auftreten. Reduzieren Sie entweder den Wert für **maxEntriesInCache** oder ändern Sie anhand der folgenden Anweisungen den Cache des Systemspeichers, um die zwischengespeicherten Daten auf den Plattenspeicher zu übertragen. Mit dieser Änderung sind sehr viel mehr parallele Sitzungen möglich.

Um zu verhindern, dass die Unica Interact-Laufzeitumgebung den verfügbaren Hauptspeicher im Heapspeicher der Java™ Virtual Machine (JVM) überschreitet, können Sie den Caching-Mechanismus so ändern, dass zum Zwischenspeichern der Daten, die den verfügbaren Hauptspeicher überschreiten, der Plattenspeicher verwendet wird.

Die Aktivierung der Plattenspeicherung führt jedoch zu einer Leistungsverschlechterung. Systemadministratoren können die den Bereitstellungssystemen verfügbare Speicherkapazität über die folgenden JVM-Parameter anpassen:

```
-Xms####m -Xmx####m -XX:MaxPermSize=256m
```

wobei die Zeichen ##### 2048 oder höher sein sollten (abhängig von ihrer Systemauslastung).



**Anmerkung:** Für Werte größer als 2048 sind normalerweise ein 64-Bit-Anwendungsserver und eine JVM erforderlich.

## Failover

Bei der Verwendung von Distributed Ignite werden die Daten über das gesamte Netz gespeichert. Wenn die Instanz, die eine Sitzung hostet, angehalten wird, gehen diese Sitzungsdaten verloren und alle nachfolgenden Anfragen für diese Sitzung schlagen fehl. Um ein Failover (Ausfallsicherung) zu erreichen, müssen Sie den Parameter numberOfBackups unter Affinium|Interact|cacheManagement|Cache Managers|Ignite|Parameter Data einstellen. Sein Wert entspricht der Anzahl der Sicherungskopien der Daten im Cache. Ein höherer Wert hat einen besseren Failover-Schutz und eine bessere

Leseleistung zu den Kosten einer niedrigeren Schreibleistung. Daher sollte er auf der Grundlage der Anwendungsfälle ausgewählt werden.

## Unica Interact-API

Verwenden Sie die Java™-Serialisierungs-API anstatt der SOAP-API. Die Serialisierungs-API ermöglicht einen besseren Durchsatz (5-10-mal mehr, je nach Anwendungskonfiguration) und eine kürzere Reaktionszeit.

Weitere Informationen zur Implementierung eines benutzerdefinierten Java-API-Zeitlimits finden Sie in der API-Dokumentation, die mit Unica Interact unter *<Interact\_home>/docs/apiJavaDoc* installiert ist, oder sehen Sie den Beispielcode und die Kommentare in *<Interact\_home>/samples/api/SampleCustomizedInteractAPI.java*.

## Installation und Netzkonfiguration

Verwenden Sie den affinen Lastausgleich und das lokale Sitzungsmanagement anstatt des verteilten Sitzungsmanagements, wenn der Unica Interact-Server die Sitzungsdaten für mehrere Unica Interact-API-Aufrufe verwalten soll und Sie zur Leistungsoptimierung den Ehcache-Cache-Manager verwenden.

Bei der Verwendung von Ehcache trägt der verteilte Modus die Kosten der Kommunikation zwischen den Unica Interact-Laufzeitservern, damit die Sitzungen konsistent bleiben. Durch lokales Sitzungsmanagement können diese Kosten vermieden werden.

Legen Sie in den Konfigurationseinstellungen von Unica Platform für die Unica Interact-Laufzeitserver die Eigenschaft `Interact > cacheManagement EHCACHE > Parameter Data > cacheType` auf den Wert `local` fest.

## Verwalten interaktiver Ablaufdiagramme

Jedes interaktive Ablaufdiagramm benötigt für die Ausführung mindestens einen Thread. Sie können ein Produktionssystem daraufhin überwachen, ob für alle interaktiven Ablaufdiagramme eine ausreichende Anzahl Threads verfügbar ist.

Überwachen Sie mithilfe der JMX-Konsole die JMX-Statistiken

`CurrentJobsInProgressBoxQueue` und `CurrentJobsInSchedulerQueue` unter `com.unicacorp.interact.flowchart`. Im Idealfall weisen diese auch unter

Spitzenbelastung den Wert Null auf, d. h., es sind ausreichend Threads vorhanden, die die Anfragen für die Ausführung der Ablaufdiagramme verarbeiten.



**Anmerkung:** Das Ausführen der JMX-Konsole wirkt sich auf die Leistung aus.

Führen Sie die JMX-Konsole daher nicht in einer Produktionsumgebung aus, es sei denn, Sie diagnostizieren ein Problem.

Steuern Sie die Warteschlangen anhand der Anzahl Threads, die von den interaktiven Ablaufdiagrammen verwendet werden. Stellen Sie die Größe des Ablaufdiagramm-Thread-Pools in Unica für die Unica Interact-Laufzeitumgebung unter `Interact > flowchart` ein.

- Setzen Sie `maxNumberOfFlowchartThreads` auf einen Wert, der mindestens die maximale Anzahl gleichzeitig angemeldeter Benutzer angibt, die am Unica Interact-Client zu erwarten sind. Wenn beispielsweise die maximale Anzahl gleichzeitig angemeldeter Benutzer 50 beträgt und bei jedem Aufruf der Segmentierung ein Ablaufdiagramm ausgeführt wird, setzen Sie den Wert für `maxNumberOfFlowchartThreads` auf 50.
- Legen Sie `maxNumberOfProcessBoxThreads` auf der Grundlage der durchschnittlichen Zahl paralleler Pfade im Ablaufdiagramm fest sowie danach, ob die Ablaufdiagramme CPU- oder E/A-gebunden sind. Der Wert muss mindestens dem Wert von `maxNumberOfFlowchartThreads` entsprechen. Wenn z. B. die durchschnittliche Zahl paralleler Pfade in den Ablaufdiagrammen 2 beträgt und alle Prozessfelder CPU-gebunden sind, setzen Sie `maxNumberOfProcessBoxThreads` auf  $2 * \text{maxNumberOfFlowchartThreads}$ . Wenn die Prozessfelder I/O-gebunden sind (z. B. wenn sie Datenbanksuchen oder Schreibvorgänge durchführen wie Prozesse "Auswahl" oder "Momentaufnahme"), muss dieser Wert eventuell höher eingestellt werden.
- Setzen Sie `minNumberOfFlowchartThreads` auf denselben Wert wie `maxNumberOfFlowchartThreads`. Setzen Sie `minNumberOfProcessBoxThreads` ebenfalls auf denselben Wert wie `maxNumberOfProcessBoxThreads`.

## Serviceoptimierung

Unica Interact verfügt über mehrere Services, die Lese- und Schreibvorgänge in Datenbanken für verschiedene Komponenten von Unica Interact verwalten, beispielsweise das integrierte Lernmodul und das Kontakt- und Antwortprotokollmodul.

Legen Sie unter `Interact > services > service name > cache > threshold` für jeden Service den Schwellenwert auf einen geeigneten Wert fest, ausgehend von der Anzahl Operationen pro Sekunde und der Zeit zum Einfügen in die Datenbank. Wenn z. B. die Anforderungen an den Systemdurchsatz bei 500 Transaktionen pro Sekunde liegen mit 2 Aufrufen mit Protokollkontakt je Transaktion, muss der Schwellenwert für `contactHist` auf einen Wert gesetzt werden, der der durchschnittlichen Zeit zum Schreiben eines Batches sowie 1000 Protokollkontakten pro Sekunde entspricht.

## Optimieren der Webanwendungsserver

In Unica Interact optimieren Sie die Webanwendung durch die Änderung von JVM-Argumenten und Verbindungen. Die JVM-Argumente wirken sich auf den Durchsatz und auf die Initialisierungszeit aus. Die von Ihnen verwendete Anzahl der Verbindungen richtet sich nach den Features, die Sie aktiviert haben.

Lesen Sie auch die Dokumentation zu Ihren Webanwendungsservern und Betriebssystemen, um Informationen zu den bewährten Verfahren für die Leistungsoptimierung zu erhalten.

### JVM-Argumente

Java™ Die Argumente der Java™ Virtual Machine (JVM) für den Webanwendungsserver werden entweder im Script des Startbefehls oder in der Administrationskonsole definiert.

- Stellen Sie sicher, dass die aktuellen Service-Packs und Patches für Ihr Betriebssystem, den Webanwendungsserver und die JVM installiert sind.
- Wenn Sie Sun HotSpot VM einsetzen, verwenden Sie das Argument `-server`, um die bestmögliche Leistung zu erreichen.
- Entscheiden Sie sich auf der Grundlage der Speicherverfügbarkeit im Server für eine maximale Heapgröße für die JVM. (Unica Interact ist keine speicherintensive Anwendung). Legen Sie die maximale und minimale Heapgröße jeweils auf denselben

Wert fest (mit den Argumenten `-Xmx` und `-Xms`). Dadurch erhöht sich zwar die Startzeit, doch der Durchsatz verbessert sich.

- Wenn die Anwendung zeitweise nicht reagiert, z. B. bei langen Reaktionszeiten von mehreren Sekunden, muss möglicherweise die Garbage-Collection-Richtlinie optimiert werden. Überwachen Sie die Ausführung der Garbage-Collection mithilfe der JMX-Konsole und überprüfen Sie die Ausgabe der Garbage-Collection, nachdem Sie die folgenden Argumente aktiviert haben:

```
-verbosegc -XX:+PrintGCDetails
```

- Tests haben ergeben, dass der Low Pause Collector die auf die Garbage-Collection zurückzuführenden langen Reaktionszeiten beseitigen konnte, ohne den Durchsatz zu beeinträchtigen. Die folgende Option hat sich für einen JVM-Heap mit 2 GB als nützlich erwiesen:

```
-XX:+UseConcMarkSweepGC -Xmn512m -XX:SurvivorRatio=6
```

Im Allgemeinen sollte die aktuelle Datensammlung etwa 1/4 bis 1/2 des gesamten Heapspeichers belegen. Der Survivor-Bereich kann auf 1/8 der Größe der aktuellen Datensammlung eingestellt werden.

- Wenn Sie bei Verwendung eines Datumsmakros (Date) eine zweistellige Jahresangabe verwenden (beispielsweise 01-01-20) oder das Datum 01/01/2020 bzw. ein späteres Datum angeben, müssen Sie beim Anwendungsstart den folgenden JVM-Parameter hinzufügen, damit die zweistellige Jahresangabe fest auf eine vierstellige Jahresangabe eingestellt wird, da dies von der Anwendung erwartet wird.

```
-DInteract.enableTwoDigitYearFix=true
```

- Unter bestimmten Umständen kann die Bereitstellung von älteren traditionellen interaktiven Kanälen oder interaktiven Kanälen mit großen Bereitstellungsverläufen das System belasten und mindestens 2048MB Java-Heapspeicher für die Campaign-Designzeit und/oder Interact-Laufzeit erfordern.

Systemadministratoren können die den Bereitstellungssystemen verfügbare Speicherkapazität über die folgenden JVM-Parameter anpassen:

`-Xms####m -Xmx####m -XX:MaxPermSize=256m`

Wobei die Zeichen #### 2048 oder höher sein sollten (abhängig von ihrer Systembelastung). Beachten Sie, dass ein 64-Bit-Anwendungsserver und eine JVM normalerweise für Werte größer als 2048 erforderlich sind.

## Referenzen

- Tuning Garbage Collection with the 5.0 Java™ Java virtual machine ([http://java.sun.com/docs/hotspot/gc5.0/gc\\_tuning\\_5.html](http://java.sun.com/docs/hotspot/gc5.0/gc_tuning_5.html))
- Java™ Java Tuning White Paper (<http://java.sun.com/performance/reference/whitepapers/tuning.html>)

## Verbindungspool

Legen Sie die Größe des Verbindungspools der Unica Interact-Laufzeitdatenquelle mithilfe der Anwendungsserverkonsole fest. Berücksichtigen Sie hierbei die Anzahl der gleichzeitig angemeldeten Benutzer und die während der Laufzeit einer Sitzung hergestellten Verbindungen, einschließlich Laden von Profilen, Laden von Angebotsunterdrückungen, Lese- und Schreibvorgängen in Ablaufdiagrammen und Lesevorgängen in Lernfunktionen.

Funktion/Option	Erforderliche Verbindungen, falls aktiviert
Mindestens eine der folgenden Funktionen ist aktiviert: <ul style="list-style-type: none"> <li>• Profiltabelle laden</li> <li>• Tabelle mit Angebotsunterdrückungen laden</li> <li>• Tabelle für Bewertungsüberschreibungen laden</li> </ul>	1 Verbindung je parallelem Clientaufruf von <code>startSession</code> oder <code>setAudience</code>  Es ist unerheblich, ob das Laden nur einer Tabelle oder aller drei Tabellen aktiviert ist.
Lernen	2 Verbindungen



Funktion/Option	Erforderliche Verbindungen, falls aktiviert
Mindestens ein Protokollierungs- oder Aufzeichnungsservice ist aktiviert	Wert von <code>Interact &gt; services &gt; threadManagement &gt; flushCacheToDB &gt; maxPoolSize</code> . Standardmäßig ist 5 eingestellt.
Ablaufdiagramme, die mindestens 1 Datenbankaufruf durchführen	Wert von <code>Interact &gt; flowchart &gt; maxNumberOfFlowchartThreads</code> . Der Standardwert ist 25.

Angenommen, die folgenden Anforderungen liegen vor:

- 30 parallele Aufrufe von `startSession` ohne Wartezeit, wenn eine Datenbankverbindung vorhanden ist (30)
- Die Lernfunktion ist eingeschaltet (2)
- Alle Services sind aktiviert (5)
- Mindestens ein Ablaufdiagramm ist implementiert, das eine Datenbankverbindung herstellt (25)
- Aktuelle Standardwerte als Grundlage (0)

Legen Sie bei diesen Anforderungen die Größe des Datenbankverbindungs-pools auf ein Minimum von 62 ( $30 + 2 + 5 + 25$ ) fest, um die optimale Leistung zu erreichen, sodass kein Nutzer der Verbindung warten muss.

## Terminologie der Optimierung

Bei der Systemoptimierung sind allgemeine Begriffe auf eine bestimmte Weise definiert.

### Antwortzeit

Die Zeit, die der Unica Interact-Laufzeitserver benötigt, um auf eine API-Anforderung zu antworten. Die Zeit wird auf der Clientseite gemessen.

### Durchsatz

Die Anzahl der Transaktionen pro Sekunde.

### **Transaktion**

Ein Aufruf an den Unica Interact-Laufzeitserver durch die Unica Interact-API, einschließlich Aufrufen, die durch die InteractAPI-Klasse definiert sind, wie z.B. `startSession` und `setAudience`. Beim Aufruf `executeBatch` handelt es sich um eine einzelne Transaktion, obwohl der Aufruf mehrere Befehle enthalten kann. Dies umfasst nicht Methoden, die mit Antwortobjekten arbeiten, z. B. die Klasse "Offer".

## Datenbankoptimierung

Die Datenbankoptimierung umfasst das Hinzufügen von Indizes zu bestimmten Tabellen sowie die Aktualisierung der Statistik.

Fügen Sie den Tabellen für Profil, Angebotsunterdrückung und Bewertungsüberschreibung geeignete Indizes hinzu.

- Profiltabellen: Erstellen Sie einen eindeutigen Index für die Felder der Zielgruppenebene.
- Tabellen für die Angebotsunterdrückung: Erstellen Sie einen Index für die Felder der Zielgruppenebene.
- Tabellen für Bewertungsüberschreibung: Erstellen Sie einen Index für die Felder der Zielgruppenebene.

Prüfen Sie, ob die Statistik zu diesen Indizes aktuell ist. Wenn z. B. die Zielgruppen-ID aus einer Kombination aus den beiden Spalten `CustomerId` und `HouseholdId` besteht, erstellen Sie in allen Tabellen einen Index für diese Spalten und aktualisieren die Statistik.

## Optimieren des ETL-Prozesses

Wenn Sie das Kontakt- und Antwortverlaufsmodul konfigurieren, wird im Hintergrund ein ETL-Prozess (ETL = Extrahieren, Transformieren, Laden) verwendet, um Daten aus den Staging-Tabellen der Laufzeitumgebung in die Unica Campaign-Kontakt- und Antwortverlaufstabellen zu verschieben.

In diesem Abschnitt werden die optionalen Konfigurationseinstellungen beschrieben, die Sie gegebenenfalls in Unica Interact ändern können, um die Leistung des Tools "ETL-Prozess" zu verbessern. Sie müssen die Standardeinstellungen der Konfigurationsparameter nicht unbedingt ändern, wenn Sie aber Änderungen vornehmen, beachten Sie die folgenden Richtlinien sowie das Unica Interact-Administratorhandbuch bei der Anpassung der Leistung des Tools.

Die im Folgenden beschriebenen Eigenschaften finden Sie in den Unica Campaign Konfigurationseigenschaften unter Campaign | Partitionen | Partition[n] | Interact | contactAndResponseHistTracking.

Konfigurationseigenschaft	Wert und Beschreibung
processSleepIntervalInMinutes	Die Anzahl von Minuten, die das Unica Interact-Modul für Kontakt- und Antwortverlauf wartet, bevor es Daten aus den Staging-Tabellen der Laufzeitumgebung von Unica Interact in die Unica Campaign-Kontakt- und Antwortverlaufstabellen kopiert. Der Standardwert ist 60.
purgeOrphanResponseThresholdInMinutes	Diese Eigenschaft legt fest, wie lange Unica Interact wartet, bis Antworten entfernt werden, die keinen entsprechenden Kontakt aufweisen (auch als "verwaiste Antworten" bezeichnet). Der Standardwert ist 180. Bei der Verarbeitung vieler Datensätze kann der Zeitraum zwischen dem Verarbeiten von Kontakten und Antworten auch größer sein. Erhöhen Sie diesen Wert, um zu verhindern, dass Antworten zu früh entfernt werden.
maxJDBCInsertBatchSize	Dies die maximale Anzahl Datensätze eines JDBC-Batches der Gesamtzahl Da-

Konfigurationseigenschaft	Wert und Beschreibung
	<p>tensätze, die das Kontakt- und Antwortprotokollmodul in einer Iteration verarbeitet, die vor dem Ausführen der Abfrage in den Unica Campaign-Systemtabellen verarbeitet (und in einem Batch zusammengestellt) wird. Der Standardwert ist 1000.</p> <p>Da dieser Wert mit der Eigenschaft <code>maxJDBCFetchBatchSize</code> zusammenhängt, müssen Sie ihn gegebenenfalls erhöhen, wenn diese Eigenschaft deutlich erhöht wird. Wenn Sie beispielsweise <code>maxJDBCFetchBatchSize</code> auf den Wert 2.500.000 setzen, können Sie für die Bearbeitung der höheren Anzahl Datensätze diesen Wert auf 10.000 festlegen.</p> <p>Beachten Sie, dass der Speicherbedarf steigt, wenn Sie diesen Wert erhöhen. Der Wert 10.000 stellt im Hinblick auf den Speicherbedarf eine geeignete Obergrenze für diese Eigenschaft dar.</p>
<code>maxJDBCFetchBatchSize</code>	<p>Legt die maximale Anzahl Datensätze fest, die für eine ETL-Stapelverarbeitung von der Staging-Datenbank abgerufen wird. Der Standardwert ist 1000. Um die Leistung des Kontakt- und Antwortprotokollmoduls zu optimieren, stellen Sie diesen Wert höher ein, als die Anzahl der Datensätze zum Kontaktverlauf, die im Allgemeinen täglich verarbeitet wird.</p>

Konfigurationseigenschaft	Wert und Beschreibung
	<p>Diese Eigenschaft wird gemeinsam mit <code>maxJDBCFetchChunkSize</code> und <code>maxJDBCInsertBatchSize</code> verwendet, um zu bestimmen, auf welche Weise die Datensätze verarbeitet werden. Angenommen, die Werte werden wie folgt festgelegt:</p> <ul style="list-style-type: none"> <li>• <code>maxJDBCFetchBatchSize</code>: 30000</li> <li>• <code>maxJDBCFetchChunkSize</code>: 1000</li> <li>• <code>maxJDBCInsertBatchSize</code>: 1000</li> </ul> <p>In diesem Beispiel werden 30.000 Datensätze abgerufen (oder bei weniger als 30.000 vorhandenen die Gesamtzahl Datensätze). Dann durchläuft das Kontakt- und Antwortprotokollmodul diese 30.000 Datensätze und verarbeitet jeweils 1.000 gleichzeitig, sodass 1.000 Datensätze in den Staging-Tabellen markiert sind und 1.000 in die detaillierte Kontaktprotokolltabelle eingefügt werden.</p>
<code>maxJDBCFetchChunkSize</code>	<p>Legt die maximale Anzahl Datensätze in einem JDBC-Datenblock fest (maximal die durch <code>maxJDBCFetchBatchSize</code> festgelegte Gesamtzahl), die in jedem Arbeitsgang verarbeitet werden sollen. Der Standardwert ist 1000. In einigen Fällen lässt sich die Leistung steigern, wenn Sie diesen Wert höher als den Eigenschaftswert von <code>MaxJDBCInsertBatchSize</code> einstellen.</p>

Konfigurationseigenschaft	Wert und Beschreibung
<code>deleteProcessedRecords</code>	<p>Diese Eigenschaft, die festlegt, ob Kontakt- und Antwortprotokolldatensätze nach der Verarbeitung beibehalten werden, ist standardmäßig auf <code>YES</code> gesetzt. Das Ändern dieses Wertes kann Ihnen mehr Kontrolle über den Datenfluss im ETL-Prozess verleihen und sich auf die Leistung auswirken (indem diese Datensätze erst zu einem späteren Zeitpunkt Ihrer Wahl entfernt werden). Sie müssen jedoch wissen, wie die manuelle Verwaltung der Datensätze gehandhabt wird, sodass diese zu geeigneten Zeitpunkten entfernt werden. Lassen Sie beim Modifizieren dieser Einstellung Vorsicht walten.</p>
<code>fetchSize</code>	<p>Das Bereitstellen eines Wertes für die JDBC-Eigenschaft <code>fetchSize</code> kann die Leistung für große Batches von Datensätzen verbessern. Die verbesserte Netzleistung fordert jedoch Zugeständnisse, da sich der größere Abrufumfang auf die Speicherbelegung auswirkt. Weitere Informationen zum Anpassen dieser Einstellung finden Sie in der Beschreibung dieser Konfigurationseigenschaft im Unica Interact-Administratorhandbuch.</p>

Ausführliche Beschreibungen der hier aufgeführten Konfigurationseigenschaften finden Sie in der Onlinehilfe zu der entsprechenden Seite "Konfiguration" oder im Unica Interact-Administratorhandbuch.

# Protokollierung

Die Protokollierungsstufe muss entweder auf INFO oder auf ERROR festgelegt sein.

Verwenden Sie in einer Produktionsumgebung niemals eine Einstellung für eine ausführliche Protokollierung, z. B. DEBUG oder TRACE.

Die Protokollierung kann in drei Bereichen konfiguriert werden:

- Legen Sie die Protokollierungsstufe in der Datei `interact_log4j.properties` fest. Standardmäßig ist diese Datei im Verzeichnis `<install_dir>/Interact/conf` installiert, wobei `<install_dir>` das übergeordnete Verzeichnis ist, in dem auch Ihre -Produkte installiert sind.
- Stellen Sie sicher, dass die Unica Interact-API keine Protokollierung durchführt. Die Protokollierung wird von einer booleschen Einstellung bestimmt, die in den Methoden `startSession` und `setDebug` verfügbar ist.
- Stellen Sie sicher, dass die JMX-Überwachung beim JMX-Vorgang `activateInfo` auf "Info" eingestellt ist.