

Intrex Professional

Intrex Compact

RELEASE 5.1



Intrex Performance

Inhaltsverzeichnis


1. Einleitung	4
2. Systemwahl	4
2.1. Systemvoraussetzungen.....	4
2.2. Speicherbedarf.....	4
2.3. Prozessoren und Taktung.....	4
2.4. Festplattengröße.....	4
2.5. Datenbank.....	4
2.5.1. Derby.....	5
2.5.2. Portal- & Datenbankserver auf einem Server.....	5
3. Portaleinstellungen	5
3.1. Speichernutzung pro Portal.....	5
3.2. Request Workerthreads.....	6
3.3. Requestqueue.....	7
3.4. Session Timeouts.....	7
4. Managereinstellungen	8
4.1. Speichernutzung des Intrexx Managers.....	8
5. Webserver	8
6. Virens Scanner	8
7. Betriebssystem	8
8. Datenbank	8
8.1. PostgreSQL.....	9
9. Browser- & Client-Optimierungen	9
9.1. Caching.....	9
9.2. Verbindungen zum Server.....	9
10. Designbedingte Optimierungen	10
10.1. Grafiken.....	10
10.2. Portalstartseite und Portlets.....	10
10.3. JavaScript.....	10
10.4. Tabellengrößen.....	11
10.5. Fremddatenquellen.....	11
10.6. Filter und Tabellen.....	12
10.7. Velocity und Prepared Queries.....	12
11. Debugging Möglichkeiten	12
12. Analysetabelle (Performance & Einstellungen)	14



Copyright






Das vorliegende Dokument ist in all seinen Teilen urheberrechtlich geschützt. Alle Rechte sind vorbehalten, insbesondere das Recht der Übersetzung, des Vortrags, der Reproduktion und der Vervielfältigung. Ungeachtet der Sorgfalt, die auf die Erstellung von Text, Abbildungen und Programmen verwendet wurde, können weder Autor, Herausgeber oder Übersetzer für mögliche Fehler und deren Folgen eine juristische Verantwortung oder irgendeine Haftung übernehmen.

Die in diesem Werk wiedergegebenen Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. können auch ohne besondere Kennzeichnung Marken sein und als solche den gesetzlichen Bestimmungen unterliegen.



Schreibkonventionen

In diesem Handbuch werden Textstellen *kursiv* dargestellt, wenn sie sich auf Einstellungen in den abgebildeten Dialogen beziehen. Menüpunkte, die in Kontextmenüs erreichbar sind, sind immer auch über das Hauptmenü erreichbar. Hauptmenüpunkte werden nicht beschrieben, es sei denn, sie sind nicht über das Kontextmenü erreichbar. Eine Beschreibung der allgemeinen Hauptmenüpunkte finden Sie im Handbuch  *Portale*. Programmiercode im Text wird in der Schriftart *Courier* dargestellt. Kontextmenüs können mit einem Klick mit der rechten Maustaste auf das beschriebene Element geöffnet werden.

<intrexx> bezeichnet im Folgenden Ihren Intrexx Installationspfad, unter Windows z.B.  `c:\intrexx\`, unter Linux z.B.  `/opt/intrexx/`. Folgende Symbole werden für die Kennzeichnung von speziellen Informationen verwendet:

-  Informationen
-  Verweise auf ein Intrexx Handbuch
-  Verzeichnisse
-  URLs
-  Klick auf Schaltflächen

Vorkenntnisse


Für das Verständnis dieser Dokumentation sind keine speziellen Vorkenntnisse erforderlich. Hilfreiche Informationen finden Sie in den Intrexx Handbüchern  *Setup* und  *Start*.

1. Einleitung

Aussagen hinsichtlich der Performance beim Arbeiten mit Intrexx sind von verschiedenen Gegebenheiten abhängig. Das folgende Dokument soll eine Hilfestellung bei der Klärung von Fragen und Problemen sein, die in Verbindung mit diesen Themen auftreten können.

2. Systemwahl

2.1. Systemvoraussetzungen

Die grundlegenden Voraussetzungen für den Betrieb eines Intrexx Xtreme Servers sind im mitgelieferten Handbuch  *Setup* zum Portal aufgeführt. Eine generelle Empfehlung für eine Konfiguration kann nicht abgegeben werden, da zu viele Faktoren einen Einfluss haben, wie beispielsweise:

- Anzahl der betriebenen Portale auf einem Server
- Anzahl der User und der durchschnittlichen Sessions (sporadische oder intensive, kontinuierliche Nutzung?)
- Art des Portals
- Geschätzter Upload-Bedarf (wie Dokumente, Bilder, Dateien) zur Bestimmung des Festplattenbedarfs

2.2. Speicherbedarf

Der Speicherbedarf ermittelt sich anhand der einzelnen, laufenden Prozesse auf dem System und wird durch das Prozessorsystem (32-Bit/64-Bit) und das passende Betriebssystem bestimmt und limitiert. Auf 32-Bit Betriebssystemen kann man bei kleinen Portalen mit 500 MB, bei größeren mit 1 GB Arbeitsspeicher pro Portal (zuzüglich Betriebssystem und Datenbank) kalkulieren. Bei 64-Bit sollte mit 750 MB bzw. 1.5 GB Speicher pro Portal gerechnet werden.

2.3. Prozessoren und Taktung

In der Regel reicht ein modernes Dual-Prozessorsystem für einen Intrexx Portalserver aus. Beim Einsatz einer Serverlizenz ist ein Dual-Prozessor oder zwei Single-Prozessoren bereits berücksichtigt. Da 32-Bit Systeme nur bis zu 4 GB RAM adressieren können, können auch 64-Bit-Prozessoren verwendet werden, um eine höhere Speichernutzung zu ermöglichen.

2.4. Festplattengröße

Für einen Intrexx Portalserver wird mindestens 400MB Festplattenplatz vorausgesetzt. Maßgebliche Faktoren für den Speicherbedarf sind die auf demselben Server installierte Datenbank sowie die zu erwartende Menge an Dateien, die über die Applikationen auf den Server geladen wird. Wird beispielsweise ein Dokumentenmanagementsystem oder ein CRM mit Ablage jeglicher Korrespondenz, eMails und Dateien mit Kunden abgelegt, kann der Speicherbedarf schnell ansteigen. Ein skalierbares System wäre in diesem Fall sinnvoll.

2.5. Datenbank

Die Wahl der Datenbank ist primär von dem im Unternehmen vorhandenen Know-How abhängig. Sekundär stellt sich die Frage der Einrichtung und Wartungsmöglichkeiten und ob z.B. eine Clusterung der Datenbank geplant ist.

2.5.1. Derby

Um für die Datenbank Derby den Speicher zu erhöhen, wechseln Sie in das Verzeichnis `<intrexx>\cfg\` und öffnen die Datei `derby.wcf`.

Entfernen Sie an folgenden Stellen das Kommentarzeichen # jeweils vor der Zeile `wrapper.java...`

```
# Initial Java Heap Size (in MB)
wrapper.java.initmemory=4

# Maximum Java Heap Size (in MB)
wrapper.java.maxmemory=128
```

Starten Sie anschließend den Derby-Dienst neu.

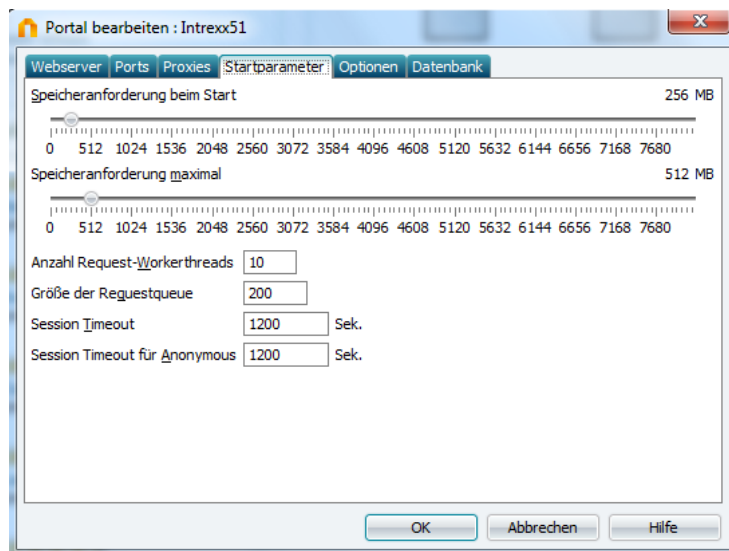
2.5.2. Portal- & Datenbankserver auf einem Server

Natürlich können Datenbank und Portalserver gemeinsam auf einem Server betrieben werden. In diesem Fall teilen sich beide die vorhandenen Ressourcen wie Prozessor, RAM und Festplatte. Dies kann unter Umständen zu Performance-Einbußen führen. Eine Trennung von Datenbankserver und Portalserver mit einer schnellen Anbindung (Gbit) ist in größeren Umgebungen obligatorisch.

3. Portaleinstellungen

3.1. Speichernutzung pro Portal

Über die Portaleigenschaften lässt sich die Größe des zugewiesenen Speichers für das Portal verändern. Melden Sie sich hierzu im Intrexx Manager am gewünschten Portal an und öffnen Sie anschließend über Eigenschaften den Einstellungsdialog.



Die hier abgebildeten Werte sollten im Normalfall als Speicheranforderungen ausreichen.

In seltenen Fällen, z.B. bei einer hohen Anzahl an eigenen eingebundenen und verwendeten Java-Klassen innerhalb eines Portals, kann es vorkommen, dass der PermGen-Speicher ebenfalls erhöht werden muss. Dies erkennen Sie daran, dass in den Log-Dateien Fehler-Einträge mit folgendem Inhalt stehen:

```
java.lang.OutOfMemoryError: PermGen space
```

Um die Speichererhöhung zu definieren, öffnen Sie die Datei portal.wcf im Verzeichnis `<intrex>\org\<portal>\internal\cfg` in einem beliebigen Texteditor.


Ändern Sie in der Zeile

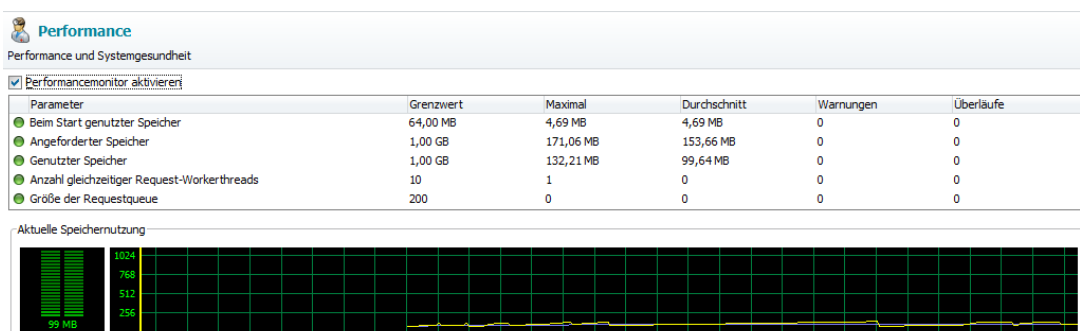
```
wrapper.java.additional.1=-XX:MaxPermSize=128m
```

den Wert von `128m` auf `256m`. Anschließend muss der Portalserver neu gestartet werden.


3.2. Request Workerthreads

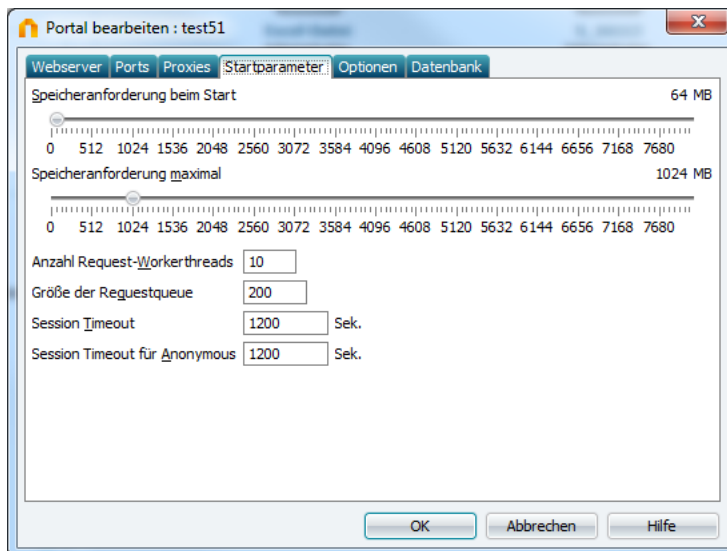
Die von den Browsern an den Portalserver gerichteten Anfragen (Requests) und die zurück übermittelten Antworten müssen entsprechend abgearbeitet werden. D.h. der Portalserver ermittelt die notwendigen Daten des Datenbankservers sowie des eigenen Dateisystems (Grafiken etc.), fügt diese in die angeforderte Seite ein und sendet die HTML-Seite an den anfragenden Browser.

Um die Anfragen zu bearbeiten, sind sogenannte *Workerthreads* auf dem Portalserver eingerichtet (*Default = 10*). Je mehr Anfragen an den Portalserver gestellt werden, desto mehr haben die Workerthreads zu tun. In seltenen Fällen reicht die Anzahl der Workerthreads nicht aus, um die Anfragemenge zu bearbeiten. In diesem Fall werden im Performance-Monitor ( *Werkzeuge/Systemmonitor/Performance*) Warnungen bzw. Überläufe angezeigt. Die Anzeige vor dem Eintrag *Anzahl gleichzeitiger Request-Workerthreads* springt von Grün auf Orange (Warnungen) bzw. Rot (Überläufe).



Sollten Warnungen oder Überläufe auftreten, kann die Anzahl der bereitgestellten Workerthreads schrittweise erhöht werden.

Wechseln Sie hierzu ins Intrex Center und klicken Sie auf  *Portaleigenschaften*. Hier wählen Sie den Reiter *Startparameter*. Nun können Sie die Anzahl der Workerthreads erhöhen. Es wird empfohlen, die Anzahl immer nur um 1 Thread zu erhöhen. Bestätigen Sie Ihre Änderungen mit *OK*.



Nach einer Änderung muss das jeweilige Portal neu gestartet werden, damit die Änderungen übernommen werden.


Beobachten Sie nun das Verhalten innerhalb der Performance-Anzeige. Sollten über einen längeren Zeitraum keine Warnungen oder Überläufe angezeigt werden, kann der aktuelle Wert beibehalten werden. Im anderen Fall erhöhen Sie den Wert erneut um 1.

3.3. Requestqueue

Die Größe der Requestqueue bestimmt die Anzahl der Anfragen, die in der Requestqueue abgelegt werden können. Die Einträge/Anfragen in der Requestqueue werden in der Reihe ihres Einganges von den Workerthreads abgearbeitet. Der Defaultwert beträgt 200 und reicht in der Regel aus. Sollten im Performance-Monitor Warnungen oder Überläufe auftreten, so kann die Größe der Requestqueue schrittweise vergrößert werden. Die Anzahl der Request-Workerthreads ist ebenfalls dafür verantwortlich, wie schnell die Requestqueue abgearbeitet wird. Sollten mehrere Threads die Anfragen bearbeiten, wird auch die gesamte Liste schneller abgearbeitet.

Zugriffszeiten auf die Datenbank und der Rücktransport der angeforderten Datensätze spielen ebenfalls eine Rolle in Bezug auf die Geschwindigkeit der Abarbeitung.


3.4. Session Timeouts

Die Zeit, nach der eine inaktive Sitzung (Verbindung zwischen Browser & Portalserver) beendet wird, kann über  *Portaleigenschaften/Startparameter/Session Timeout* (in Sekunden) eingestellt werden.

Neben Sicherheitsaspekten ist auch die Performance ein Grund für eine moderat gewählte Sitzungsdauer. Die Standardeinstellung beträgt 1200 Sekunden, also 20 Minuten, was in der Regel ausreichend sein sollte. Arbeitet ein Benutzer kontinuierlich in einer Applikation, wird der Timeout ständig zurückgesetzt und das automatische Logout tritt nicht in Aktion. Falls doch, kann das Logout durch den Benutzer abgebrochen werden bzw. kann er durch ein erneutes Login zum letzten aktiven Zustand gelangen.

4. Managereinstellungen

4.1. Speichernutzung des Intrexx Managers

Gehen Sie wie folgt vor, um den verfügbaren Speicher für den Intrexx Portal Manager zu erhöhen. Navigieren Sie in das Verzeichnis  `<intrexx>\client\bin\`. Wechseln Sie hier in den Ordner des zugehörigen Betriebssystems. Hier finden Sie je nach Betriebssystem die Datei `manager.bat` bzw. `manager.sh`. Legen Sie von der unveränderten Datei ein Backup an und öffnen Sie sie anschließend mit einem beliebigen Texteditor.

Suchen Sie nach dem Eintrag `-Xmx256m` und ändern Sie diesen auf `-Xmx512m` ab. Starten Sie anschließend Supervisor- und Portaldienst neu.

5. Webserver

Der verwendete Webserver kann ebenfalls hinsichtlich der Performance optimiert werden. Eine Möglichkeit ist, die HTTP-Komprimierung für statische Dateien am Webserver zu aktivieren. Komprimiert werden sollten `*.js` und `*.css` Dateien (auch beim IIS muss dies noch extra eingestellt werden).

Im Folgenden finden Sie Anleitungen zum Aktivieren der HTTP-Komprimierung für Internet Information Services (ISS)-Server und für Apache.

 Anleitung IIS 6: http://www.devtrain.de/artikel_902.aspx

 Anleitung Apache: http://httpd.apache.org/docs-2.0/mod/mod_deflate.html

6. Virens Scanner

Prüfen Sie, falls ein Virens Scanner läuft, wie sich die Performance bei deaktiviertem Virens Scanner verhält. Es ist nicht notwendig, dass der Virens Scanner das komplette Intrexx Verzeichnis scannt. Lediglich folgende Pfade sind für einen Scan relevant, da es Benutzern nur in diesen beiden Verzeichnissen möglich ist, Dateien in ein Portal hochzuladen.

`<intrexx>\org\portal\internal\files`

`<intrexx>\org\portal\external\htmlroot\userfiles`

 **Hinweis für Benutzer von McAfee VirusScan:**

Beim Surfen mit dem Internet Explorer scannt McAfee im Browser alle JavaScript-Ausführungen. Hierdurch verlangsamt sich ein Seitenaufbau deutlich.

7. Betriebssystem

Auf der Betriebssystemebene geht es in erster Linie um die Überwachung der Prozessor- und der Speicherauslastung. Bei Performanceproblemen sollte hier ebenfalls ein Blick auf das Systemverhalten (im Taskmanager) geworfen werden. Sollten die Prozessoren kontinuierlich ausgelastet sein, könnte ein Ausbau um weitere Prozessoren notwendig sein.

8. Datenbank

Grundsätzlich sollten sich die verwendete Datenbank und deren Version auch im Einsatz der adäquaten Treiber in Intrexx widerspiegeln. Prüfen Sie daher den Versionsstand der installierten JDBC-Treiber in der verwendeten Intrexx Version und aktualisieren Sie diese gegebenenfalls mit Treibern der aktuell installierten Datenbank-Version.

8.1. PostgreSQL

Ein verzögertes Antwortverhalten sowohl in Applikationen als auch bei einer Suche kann zu einer CPU-Auslastung von 90-100% führen. Die Ursache hierfür liegt an der verwendeten Datenbank und deren routinemäßigen Pflege. In PostgreSQL sollte deshalb aus mehreren Gründen regelmäßig der Befehl *VACUUM* ausgeführt werden, um:

- den von aktualisierten oder gelöschten Zeilen belegten Speicherplatz wiederzugewinnen.
- die vom PostgreSQL Anfrageplaner verwendeten Datenstatistiken zu aktualisieren.
- sich gegen den Verlust alter Daten durch Überlauf der Transaktionsnummern zu schützen.

 Dies sollte nicht im laufenden Betrieb vollzogen werden.

Seit PostgreSQL 8.1 besteht die Möglichkeit, diese Operation durch eine Auto-Vacuum-Demon ausführen zu lassen. Dieser kann so konfiguriert werden, dass der *VACUUM*-Befehl in regelmäßigen Abständen automatisch aufgerufen wird. Die Befehlsausführung muss über die Konfigurationsdatei *postgresql.conf* aktiviert werden.

Weitere Informationen finden Sie unter

 <http://www.postgresql.org/docs/8.1/static/maintenance.html>

 <http://www.postgresql.org/docs/8.1/static>

 <http://www.postgresql.org/docs/9.0/static/maintenance.html>

 <http://www.postgresql.org/docs/9.0/static>

9. Browser- & Client-Optimierungen

Auch auf Browser- & Clientseite gibt es verschiedene Möglichkeiten, Optimierungen vorzunehmen. Bei diesen Optimierungen müssen in der eigentlichen Applikation keine Änderungen vorgenommen werden.

9.1. Caching

Auf der Seite des Browser besteht eine Möglichkeit darin, das browserseitige Caching zu aktivieren. Die Browser erkennen nur sehr eingeschränkt Änderungen an dynamisch generierten Inhalten und erfordern deshalb eine regelmäßige Aktualisierung des Caches, damit z.B. eine veränderte JavaScript-Funktion auch verwendet wird und nicht fälschlicherweise eine alte Version, die sich noch im Cache befindet. Hierzu können im Intrexx Manager JavaScript- und Velocity-Aufrufe explizit veröffentlicht werden. Hiermit erhält der Browser die Aufforderung, die Dateien neu zu laden und im Cache zu erneuern. Wählen Sie im Modul *Applikationen* den Menüpunkt *Datei / JavaScript/Velocity Dateien veröffentlichen*.

9.2. Verbindungen zum Server

Die Microsoft Windows Internet Application (WinInet) beschränkt die Anzahl gleichzeitiger Verbindungen, die von einem Client zu einem HTTP-Server aufgebaut werden können. Wird diese Höchstgrenze überschritten, werden die Anfragen so lange blockiert, bis eine der aktuellen Verbindungen abgeschlossen ist.

Es handelt sich hierbei um ein beabsichtigtes Verhalten, das mit HTTP-Spezifikations- und Industriestandards übereinstimmt. Die Auswirkung dieser Beschränkung auf Ihre

Anwendung besteht darin, dass das Ausführen von Aufrufen wie *HttpSendRequest* und *InternetOpenURL* augenscheinlich mehr Zeit benötigt, da sie warten, bis vorherige Verbindungen freigegeben werden. Dies kann bei Ajax-Requests eine Rolle spielen.

Zu diesem Thema finden Sie einen Microsoft Knowledgebase-Eintrag (KB83110) unter

 <http://support.microsoft.com/kb/183110/EN-US/>

Sollten Sie diese Einstellungen vornehmen, beachten Sie bitte, dass sich die getätigten Einstellungen nur auf den aktuell angemeldeten Nutzer beziehen. Bei anderen auf dem Rechner angelegten Benutzerprofilen wirken sich die Einstellungen nicht aus.

10. Designbedingte Optimierungen

10.1. Grafiken

Verwenden Sie für das Web optimierte Grafiken und versuchen Sie, die Anzahl eingebundener Grafiken gering zu halten.

Versuchen Sie mit kleinen Grafiken und deren Kachelung zu arbeiten, um größere Flächen zu füllen. Als Grafikformat ist PNG bevorzugt zu verwenden, da dieses Format trotz höherer Farbzahl kleinere Dateigrößen als JPG und GIF besitzt. (ein Beispiel: 110kB JPG, 37kB GIF, 31kB PNG bei vergleichbarer Bildqualität).

10.2. Portalstartseite und Portlets

Überprüfen Sie die Anzahl der Portlets auf der Portalseite. Jedes Portlet verlangsamt den Zugriff. Portlets, die Informationen aus dem Internet beziehen, können ebenfalls eine Bremse darstellen, wenn die Kommunikation ins Internet oder die Antwort mit Verzögerungen behaftet ist.

Beim Laden Ihrer Startseite wird standardmäßig das Hauptportal mitgeladen. Um dies beispielsweise für anonyme Nutzer zu vermeiden, können Sie stattdessen eine statische HTML-Seite laden, die als Startseite fungiert.

Kopieren Sie hierzu Ihre erstellte HTML-Seite in das Verzeichnis `<intrex>/org <portalname>/external/htmlroot`. Öffnen Sie danach im Portal designer unter *Extras* den Dialog *Menüstruktur bearbeiten*. Wählen Sie den Menüpunkt *Home*. Auf der rechten Seite können Sie anschließend unter *Startseite des Menüpunkts* den Namen ihrer statischen HTML-Datei eintragen.

Falls Sie nur für anonyme Benutzer eine statische Datei verwenden wollen, wählen Sie die Benutzerrechte so, dass bereits angelegte Benutzer das Recht haben, auf den Menüpunkt *Home* (eingestellt auf *Portalseite*) zuzugreifen. Legen Sie zusätzlich einen weiteren Menüpunkt an, der ebenfalls *Home* lautet und dessen Startseite die von Ihnen erstellte statische HTML-Seite ist. Auf diesen zweiten Menüpunkt setzen Sie die Benutzerrechte so, dass nur anonyme Benutzer darauf zugreifen dürfen.

Falls noch nicht geschehen, müssen Sie beide Home-Menüpunkte im Fenster *Menüstruktur bearbeiten* an die oberste Stelle verschieben. Dies muss geschehen, da immer der erste Menüpunkt geladen wird, zu dem ein Benutzer zugewiesene Rechte besitzt.

10.3. JavaScript

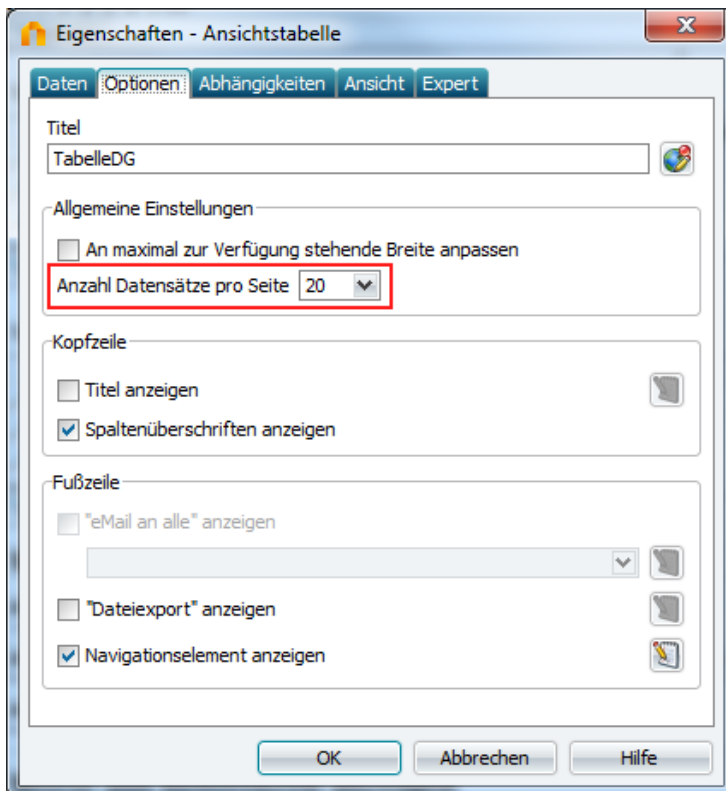
Beeinträchtigungen können auch durch selbst verfasste und eingebundene JavaScript-Funktionen entstehen. Schleifenstrukturen und Timer sind hierbei die häufigsten Verur-

sacher. In Bezug auf JavaScript-Funktionen einzelner Portlets können über die zu Grunde liegende Ansichtseite und deren *onload()*-Ereignis Funktionen ausgeführt werden, die Verzögerungen oder Fehlverhalten verursachen können. Gleiches gilt auch innerhalb von Applikationen und deren Ansichts- und Eingabeseiten.

Prüfen Sie in diesem Zusammenhang auch die JavaScript-Datei *custom.js* im Verzeichnis `<intrexx>/org/<portalname>/external/htmlroot/include/custom`. Diese Datei wird automatisch in jede Seite eingebunden. Sollten sich hier Fehler befinden, würde sich diese auf jede geladene Seite auswirken.

10.4. Tabellengrößen

Für Ansichtstabellen kann im Applikationsdesigner eine Anzahl von Zeilen vorgegeben werden, die zu Beginn und während des Blätterns durch die Tabelle dargestellt werden. Wird die Anzahl der Zeilen sehr hoch gesetzt, erhöhen sich auch die Ladezeiten, da diese Menge an Daten ermittelt, übermittelt, gerendert und vom Browser interpretiert werden muss. Halten Sie die Anzahl der Zeilen daher möglichst gering – besonders in der Default-Einstellung.



10.5. Fremddatenquellen

Verwendete Datenquellen, die per Fremddatenintegration in Intrexx eingebunden werden, können den Seitenaufbau und dessen Geschwindigkeit beeinflussen. Enthält die Datenquelle eine große Anzahl von Datensätzen und Spalten, kann die Datenermittlung längere Zeit benötigen. Aber auch fehlerhafte SQL-Statements in angelegten Views können ein Grund für Performance-Einbrüche sein. Prüfen Sie daher die Fremddatenzugriffe auf Datenbankebene. Hierzu können Sie sich über das Expert-Attribut *showQuery* (*Boolean, true*) das für die jeweilige Ansichtstabelle resultierende SQL-Statement ausgeben lassen. Dieses kopieren Sie in ihr Datenbank-Administrationswerkzeug wie z.B. den Query Analyzer des Microsoft Enterprise Managers, das Abfragewerkzeug von pgAdmin der PostgreSQL-Datenbank oder Oracle S*Plus. Überprüfen Sie dort das Verhalten des SQL auf Performance und evtl. auftretende Fehler.

Ein fremdes Datensystem kann durch die eigentliche Software bereits stark belastet sein. Zu einer evtl. hohen Anzahl an Zugriffen auf das System durch die eigene Software kommen zusätzlich die Zugriffe von Intrexx. Verzögerungen in der Antwortzeit wirken sich dann in beiden Systemen aus. Prüfen Sie die Belastung des Systems daher auch aus diesem Blickwinkel.

10.6. Filter und Tabellen

Statische Filter auf Ansichtstabellen können das übertragene Volumen an Daten reduzieren. Sollen Daten nur ab einem bestimmten Zeitpunkt verwendet werden, kann durch ein Ausfiltern von Daten z.B. vor dem jeweils aktuellen Jahr ggf. eine Optimierung des Antwortverhaltens erzielt werden.

Beachten Sie, dass dynamische Filter erst nach Angabe des Suchkriteriums mit der Ausführung der Filterung beginnen. Bei, bzw. vor dem Laden einer Seite, werden diese Filter nicht aktiv. Es wird somit die volle Ergebnismenge abgerufen (auch wenn diese nicht angezeigt werden!).

Arbeiten Sie also, wenn möglich, mit statischen Filtern, um die Ergebnismenge bereits von vorne herein zu reduzieren.

10.7. Velocity und Prepared Queries

Der Einsatz von Velocity-Templates mit sogenannten Prepared Queries zur Ermittlung von Daten aus Fremddatengruppen kann unter Umständen eine schnellere Alternative zu Ansichtstabellen darstellen. Das Prepared Query wendet ein SQL-Statement serverseitig an und verarbeitet die Ergebnisse ebenfalls auf dem Server. Sollen Daten abhängig von einem Vorgabewert ermittelt werden, so empfiehlt sich eine Assistenten-Konstruktion der Applikation. Auf der ersten Seite wird der Vorgabewert (z.B. Primary Key) erfasst und als Request-Parameter an die zweite Seite übergeben. Dort kann der Request-Parameter in der WHERE-Klausel des SQL-Statements verarbeitet werden.

Diese Verfahrensweise wird jedoch nur bei extremen Anforderungen Verwendung finden. Hierbei können Sie auch erfahrene Consultants der Firma United Planet unterstützen.

11. Debugging Möglichkeiten

Intrexx unterstützt die Java Management Extensions (JMX). Standardmäßig ist JMX allerdings aus Sicherheitsgründen deaktiviert. Auf Grund der nicht vorhandenen Benutzer-Authentifizierung sollte das folgende Konfigurationsbeispiel nicht dauerhaft in Produktionsumgebungen eingesetzt werden. Auf Dauer sollte mit einem passwortgeschützten und verschlüsselten Zugang gearbeitet werden. Wie dieser einzurichten ist, kann der einschlägigen JMX-Dokumentation entnommen werden.

Um einen offenen Zugang zu konfigurieren, kann man in der Service-Wrapper-Konfigurationsdatei (im Falle eines Portalserver `<intrexx>\org\<portal>\internal\cfg\portal.wcf`) folgende Einträge hinzufügen:

```
wrapper.java.additional.20=-Dcom.sun.management.jmxremote.port=9001  
wrapper.java.additional.21=-Dcom.sun.management.jmxremote.authenticate=false  
wrapper.java.additional.22=-Dcom.sun.management.jmxremote.ssl=false
```

Die laufenden Nummern hinter `wrapper.java.additional` müssen gegebenenfalls so angepasst werden, dass sie eindeutig sind. Als Port muss natürlich ein freier Port gewählt werden.

Hat die Einrichtung funktioniert, sollte der Server ohne Probleme starten (siehe Logdateien) und mit jedem JMX-Client erreichbar sein. Der bei den Sun-JDK-Tools mitgelieferte Client kann mit `jconsole servername:port` gestartet werden.

Dokumentation zur Java Console finden Sie unter:

 <http://java.sun.com/j2se/1.5.0/docs/guide/management/jconsole.html>

12. Analysetabelle (Performance & Einstellungen)

Portal						
Anzahl der User						
Anzahl der Sessions (Spitzenwert)						
Parameter	Def.	Werte 1	Werte 2	Werte 3	Werte 4	Werte 5
Speicheranforderung bei Start (MB)	64					
Speicheranforderung maximal (MB)	256					
Anzahl der Request-Workerthreads	10					
Größe der Requestqueue	200					
Session-Timeout	1200					
Dynamisches Pooling	Ja					
Performance-Monitor						
Beim Start genutzter Speicher Maximal						
Beim Start genutzter Speicher Warnungen						
Beim Start genutzter Speicher Überläufe						
Angeforderter Speicher Maximal						
Angeforderter Speicher Warnungen						
Angeforderter Speicher Überläufe						
Genutzter Speicher Maximal						
Genutzter Speicher Warnungen						
Genutzter Speicher Überläufe						
Request-Workerthreads Maximal						
Request-Workerthreads Warnungen						
Request-Workerthreads Überläufe						
Requestqueue Maximal						
Requestqueue Warnungen						
Requestqueue Überläufe						
Verbindungen (statischer Pool)						
Verbindungen (dynamischer Pool)						
Feststellungen Default						
Feststellungen Werte 1						
Feststellungen Werte 2						
Feststellungen Werte 3						
Feststellungen Werte 4						
Feststellungen Werte 5						
Datum		Bearbeiter				

Die Analysetabelle kann idealerweise auch als Informationsquelle für den Intrexx Support verwendet werden. Bei Performance-Problemen sollte diese ausgefüllt und zusammen mit allen LOG-Dateien des Portals (Im Portalmanager erreichbar über *Werkzeuge/System Care/Diagnose* dem Support zugesandt werden.