



PNMsoft Knowledge Base

Sequence User Guides

Background Runtime Service Overview

Sept. 2014

Product Version 7.0 and above

© 2014 PNMsoft All Rights Reserved

This document, including any supporting materials, is owned by PNMsoft Ltd and/or its affiliates and is for the sole use of the PNMsoft customers, PNMsoft official business partners, or other authorized recipients. This document may contain information that is confidential, proprietary or otherwise legally protected, and it may not be further copied, distributed or publicly displayed without the express written permission of PNMsoft Ltd. or its affiliates.

PNMsoft UK 38 Clarendon Road Watford Hertfordshire WD17 1JJ

Tel: +44(0)192 381 3420 • Email: info@pnmsoft.com • Website: www.pnmsoft.com

Microsoft Partner

Gold Application Development

TABLE OF CONTENTS

General Document Information	1
Purpose.....	1
Prerequisites.....	1
Introduction	2
BRS Installation Topologies	2
Installation and BRS Config File	2
Configuration Options	3
Redirector Thread	3
EmailSender Thread.....	3
dbListener Thread.....	5
StatisticsCalculator Thread	5
Subscriber Thread.....	5
ExpiredTemporaryAttachmentsCleanerThread.....	6
ClosedInstancesSeperator Thread	7
ClosedRecipientBreadCrumbsCleaner Thread	7
ReminderMessageCreator Thread	8
EmailSenders Thread	8
Messaging Thread.....	9
Shutdown Configuration	11
Multiple BRS Installation.....	11
Troubleshooting	12

General Document Information

Purpose

The purpose of this document is to describe how to configure the Background Runtime Service (BRS).

Prerequisites

- You should have access to the server on which the BRS is installed.

Introduction

The BRS (Background Runtime Service) is an integral part of the functionality of Sequence. The BRS is installed as part of the Sequence installation, and continues to run in the background at all times.

The BRS is a Windows service which has several threads. Each of these threads is responsible for performing one of the following actions:

- Redirecting workflows
- Sending emails
- Executing the Database Listener activity
- Synching work with other BRS instances
- Calculating statistics
- Moving closed process instances to separate tables
- Creating reminders
- Cleaning the table of temporary attachments

Each of the above threads includes configuration options, such as how often and when to perform the action. The configuration options are described in the following sections.

BRS Installation Topologies

There can be several instances of the BRS. Each one should be installed on a dedicated server. The BRS instances all work together, dividing their work on a load balancing basis.

The registration of a new BRS instance is done automatically. The system is updated automatically when services are shut down or go offline, without any need for manual action.

Installation and BRS Config File

When you install Sequence, you can select the location of the BRS config file. This file is named: *PNMsoft.Sequence.WindowsServices.BRS.exe.config*

To configure BRS options, you can edit this file, as described below. You may also choose to use the default options which are set during the installation, if no special functionality or optimization is required.

Configuration Options

The BRS config file `PNMsoft.Sequence.WindowsServices.BRS.exe.config` contains several sections for each of the threads. Each section has several options which you can edit.

Redirector Thread

The redirector thread redirects workflows, progressing workflows to subsequent activities in the flow. The redirector picks workflow instances that have an open link with type: Automatic or All, and whose last updated or next redirect date matches the last redirect of the BRS instance. The workflows are redirected in a FIFO (first in first out) order. This thread is defined in this section:

```
<redirectorThread  
  
    type="PNMsoft.Sequence.WindowsServices.BRS.BuiltInThreads.Redirector,  
PNMsoft.Sequence.WindowsServices.BRS"  
  
    interval="10"  
  
    pageSize="0"  
  
>
```

Options

interval: The idle time of the redirector thread (in minutes) between completing a cycle and starting a new cycle. For example, if set to 10, the thread will wait 10 minutes between cycles. To disable the thread execution set its interval to 0. (Unit = minutes).

pageSize: You can use this option to redirect chunks of the workflow, especially if there are a large number of instances and you are interested in improving performance. For example, if set to 15, the thread will redirect 15 workflow instances at a time. If set to 0, the thread will redirect all workflow instances at once. (Unit = number of workflow instances).

Note: If there is more than one BRS installed, this thread will pick only the relevant part of the work.

EmailSender Thread

This thread defines configuration for emails that are sent from Sequence. Emails can be sent for messages, tasks and other notifications. This thread is defined in this section:

```
<emailSenderThread  
  
    type="PNMsoft.Sequence.WindowsServices.BRS.BuiltInThreads.EmailSender,  
PNMsoft.Sequence.WindowsServices.BRS"  
  
    interval="10"  
  
    sendRetries="10"  
  
    siteUrl="http://serverName/Sequence/"  
  
    outlookWebService="http://serverName/Interfaces/"  
>
```

```
messageXsltFolder="E:\Projects\SequenceEngine\PNMsoft.Sequence\PNMsoft.Sequence.WindowsServices.BRS\Messaging\XSLT"

messageLogosFolder="E:\Projects\SequenceEngine\PNMsoft.Sequence\PNMsoft.Sequence.WindowsServices.BRS\Messaging\Images"

sendLogos="True"

smtpServer="SMTPServer"

smtpPort="25"

enableSSL="False"

useDefaultCredentials="True"

username=

password=

isHtmlBody="True"

attachmentLimitationSize="2048"

/>
```

Options

interval: The idle time of the thread between sending emails. (Unit = minutes).

sendRetries: How many times to try sending an email.

siteURL: The location of the Flowtime (runtime) site. This is provided as a link within each Sequence email, enabling users to easily return to the Flowtime site.

messageXsltFolder: The path to the message body XSLT folder in which the email body format is defined. This folder contains email message body templates per language.

messageLogosFolder: The path to the folder that contains the logo files for Sequence emails.

sendLogos: Defines whether or not to include logos as part of Sequence emails. (Values: True/False).

smtpServer: The SMTP server via which Sequence sends emails.

smtpPort: The SMTP port – Use this property if your SMTP is not communicating on port 25.

enableSSL: Enable sending using SSL protocol. (Values: True/False). The default value is False.

useDefaultCredentials: If you work with SSL and set this to “True”, Sequence will send a request to SMTP with the user that runs the BRS. If set to “False”, then you must define the **username** and **password** fields with the Domain\username and password with which to authenticate against the SMTP server. Note: the password is encrypted.

useHtmlBody: Determines whether the email body is built using HTML or plain text (Values: True/False). Set this option to false if your users cannot see HTML emails with their email client.

attachmentLimitationSize: Defines the maximum size of an attachment to an email. (Unit: Kb).

dbListener Thread

This thread executes DB Listener activities, meaning that it checks for changes in the database table (defined in the activity), and initiates a workflow if the DB Listener activity is at the beginning of the workflow and a new record was added. If the DB listener is in the middle of the workflow the process will resume if the record with the matching key has changed. ([Click here](#) for more details on the DB Listener activity).

This thread is defined in this section:

```
<dbListenerThread  
  
    type="PNMsoft.Sequence.WindowsServices.BRS.BuiltInThreads.DatabaseListener,  
PNMsoft.Sequence.WindowsServices.BRS"  
  
    interval="10"  
  
>
```

Options

interval: The idle time of the thread between synchronizing the DB Listener activities. (Unit = minutes).

StatisticsCalculator Thread

This thread calculates and updates the value of statistics in Sequence workflows. ([Click here](#) for more details on defining Statistics). This thread is defined in this section:

```
<statisticsCalculatorThread  
  
    type="PNMsoft.Sequence.WindowsServices.BRS.BuiltInThreads.StatisticsCalculator,  
PNMsoft.Sequence.WindowsServices.BRS"  
  
    interval="10"  
  
>
```

Options

interval: The idle time of the thread between calculating statistics. (Unit = minutes).

Subscriber Thread

This thread enables synchronization of multi BRS instances. By registering their status to the tblOnlineBRSInstances table (as described in the Introduction above) this thread enables the instances to redistribute their work on workflow instances for redirect and move to close. This thread is defined in this section:

```
<subscriberThread
```

```
type="PNMsoft.Sequence.WindowsServices.BRS.BuiltInThreads.Subscriber,  
PNMsoft.Sequence.WindowsServices.BRS"
```

```
id="2F773C63-0D3C-46d5-A047-30E8E341B865"
```

```
timeout="20"
```

```
/>
```

Options

id: The ID of the BRS instance.

Note: If you have more than one BRS installed, this key must be unique per BRS.

timeout: If a BRS instance exceeds the timeout value (e.g. 20 minutes), then the BRS concludes that this BRS instance is not active, and it redistributes work among the other BRS instances. Also, when reaching half this time (e.g. 10 minutes), the BRS instance notifies the other BRS instances that it is active. (Unit = minutes).

ExpiredTemporaryAttachmentsCleanerThread

When a user uploads an attachment to a form (even before hitting Submit), a temporary version of the attachment file is saved to the tblAttachmentsTemp table in the Database. This thread deletes these temporary attachments from this table.

This thread is defined in this section:

```
<expiredTemporaryAttachmentsCleanerThread
```

```
type="PNMsoft.Sequence.WindowsServices.BRS.BuiltInThreads.ExpiredTemporaryAtt  
achmentsCleaner, PNMsoft.Sequence.WindowsServices.BRS"
```

```
interval="60"
```

```
expirationPeriod="04:00"
```

```
/>
```

Options

interval: The idle time of the thread between cleaning up temporary attachments. (Unit = minutes).

expirationPeriod: the time after which the BRS deletes a temporary attachment on a form (Unit = hours:minutes).

ClosedInstancesSeperator Thread

This thread goes over all the closed workflow instances and moves all their metadata records to tables such as tblInstanceWorkflowsClosed, tblInstanceActivitiesClosed, etc. This thread is defined in this section:

```
<closedInstancesSeperator  
  
    type="PNMsoft.Sequence.WindowsServices.BRS.BuiltinTherads.ClosedInstancesSeperator, PNMsoft.Sequence.WindowsServices.BRS"  
  
    interval="10"  
  
    pageSize="100"  
  
    startAt="02:00"  
  
    stopAt="04:00"  
  
>
```

Options

interval: The idle time of the thread between checking for and moving closed workflow instance content. (Unit = minutes).

pageSize: This option enables the thread to work in chunks, handling x number of closed workflow instances at a time. This is relevant if you have a large number of workflow instances and want to optimize performance. (Unit: number of workflow instances).

startAt: The time of day the thread begins to work.

stopAt: The time of day the thread ends its work.

If you want the thread to work all the time, leave the startAt and stopAt options empty.

Note: It is recommended to set this thread to run at times when the system is less in use.

ClosedRecipientBreadCrumbsCleaner Thread

Sequence saves breadcrumbs to track add recipient, remove recipient, and replace recipient actions on workflow tasks and messages. These breadcrumbs are saved to Database tables. This thread cleans up the tblRecipientsBreadcrumbs and the tblReminderBreadcrumbs tables for closed messages/tasks.

This thread is defined in this section:

```
<closedRecipientBreadCrumbsCleanerThread  
  
    type="PNMsoft.Sequence.WindowsServices.BRS.BuiltinTherads.ClosedRecipientBreadCrumbsCleaner, PNMsoft.Sequence.WindowsServices.BRS"  
  
    interval="120"  
  
>
```

Options

interval: The idle time of the thread between cleaning up the breadcrumbs tables. (Unit = minutes).

ReminderMessageCreator Thread

This thread creates reminder messages as defined in the message wizard or the message instance ([click here](#) for more information about reminders). This thread is defined in this section:

```
<reminderMessageCreator
    type="PNMsoft.Sequence.WindowsServices.BRS.BuiltInTherads.ReminderMessageCreat
or, PNMsoft.Sequence.WindowsServices.BRS"
    interval="10"
/>
```

Options

interval: The idle time of the thread between creating reminders. (Unit = minutes).

EmailSenders Thread

This thread defines from which email Sequence messages are sent. This thread is defined in this section:

```
<emailSenders>
    <defaultEmailSender email="sequence@pnmsoft.com" />
    <!-- Example how to add e-mail sender
        <workflowEmailSenders>
            <add workflowId="" email=""></add>
        </workflowEmailSenders>
    -->
</emailSenders>
```

Options

defaultEmailSender email: define the default email sender for Sequence emails.

You can also add email senders per workflow using the example above (in green).

Messaging Thread

In Sequence there are several different types of messages: Normal, Rollback, Answer, Reminder, Comment, Delegation, and Question. Sequence enables you to configure the sender mode for these messages.

The rules defining which sender mode is used for messages are as follows:

- You can set the sender mode in the **Message wizard > More Options** tab within the App Studio. There are several options for sender mode in that tab, but if you choose “System Default”, then the sender mode is set according to what is defined in the BRS config file.
- In the BRS config file (see the section immediately below), if the sender mode is set to “Predefined”, then messages are sent according to the <emailSenders> section of the BRS config file (see that section above).
- Other options you can enter here are “WorkflowInstanceCreator” / “ActivityInstanceCreator” / “MessageSender”.

The messaging thread is defined in this section:

```
<messaging>

  <messageTypes>

    <!-- workflow message -->

    <add id="00000000-0000-0000-0000-000000000000" senderMode="Predefined" />

    <!-- rollback message -->

    <add id="E993B9F3-3A3F-4DE6-B38B-A086A1729CA1" senderMode="Predefined" />

    <!-- answer message -->

    <add id="67244068-B77E-410D-B26D-E34A607514FD" senderMode="Predefined" />

    <!-- comment message -->

    <add id="F883F524-2707-4959-8713-08B4D504B238" senderMode="Predefined" />

    <!-- delegation message -->

    <add id="100215E0-AB4C-4D11-BB7C-0FC313248895" senderMode="Predefined" />

    <!-- question message -->

    <add id="E966270A-8759-45CD-B222-3CAB5A7F0DB0" senderMode="Predefined" />

  </messageTypes>

</messaging>
```

Options

id : the message type id as detailed in the tblMessageBehaviorTypes table.

senderMode: Select the sender mode value for each message type. (Values: “Predefined”, “WorkflowInstanceCreator”, “ActivityInstanceCreator”, “MessageSender”).

Shutdown Configuration

Sometimes the BRS may be shut down while Sequence processes are in execution. You can configure how long the Sequence engine will wait after a BRS shut down, before shutting itself down. You can do so by adding the following configuration section to the BRS.config file:

```
<configSections>
...
    <section name="sequence.host"
type="PNMsoft.Sequence.Configuration.WorkflowEngineHostConfigurationSection,
PNMsoft.Sequence, Version=7.0.0.0, Culture=neutral, PublicKeyToken=0a1a1b90c1c5dca1"
allowDefinition="MachineToApplication" />
...
</configSections>

...
<sequence.host shutdownTimeout= "00:00:30" />
...
```

Options

sequence.host shutdownTimeout: The amount of time in hrs:min:sec that the Sequence engines waits after a BRS shutdown, before shutting itself down. For example, if the BRS shuts down and this option is set to 00:01:30, then the Sequence engine will wait 1:30 min. before shutting itself down. If the BRS is turned on again during that time, the Sequence engine will continue to run.

If nothing is configured, the default will be 30 seconds timeout to complete the current execution.

Multiple BRS Installation

It's possible to have multiple BRS instances on different servers that will work on the same Sequence environment, and distribute the work between them.

To do this, please follow the steps below:

1. Run a full Sequence installation on the first server, BRS included (including Email Listener).
2. Run a second Sequence installation on the second server:
 - a. Choose to install only the BRS component (including Email Listener).
 - b. Enter the database connection details so it connects to the same database as the first BRS instance.
 - c. Enter BRS user that exists in tblEmployees.
3. Once installation is complete, start the BRS service

Troubleshooting

The following are some troubleshooting techniques for common BRS scenarios:

BRS Stops

If the BRS stops unexpectedly it might indicate a host environment issue. We recommend looking at the server logs to identify the root cause.

BRS Won't Start

If you try to start the BRS service and it doesn't start, there may be two reasons:

1. The user that is defined to run the BRS during installation doesn't have permissions to run windows services.
2. The BRS can't connect to the Sequence Database using the connection string you provided during installation. Either the connection string is wrong, or there is a problem with the Database server.
3. The user that is defined to run the BRS has changed his/her password.

Security Exception in Reminder Thread

If you have received a security exception in the redirect or reminder thread, check that the user is a valid Sequence user.

Email Sending Issues

To troubleshoot email sending failures, check that the SMTP port (25) is open on the server, and check the SMTP server configuration. Another common mistake is that Emails are not defined for the activity.