

## [Tech Note 930](#)

# Wonderware Application Server Clean-up Guide

---

All Tech Notes, Tech Alerts and KBCD documents and software are provided "as is" without warranty of any kind. See the [Terms of Use](#) for more information.

Topic#: 002746

Created: February 2013

Updated: August 2013

## Introduction

When working with a Galaxy, some of the Galaxy-related files can become quite large, impacting system performance. Investigation and continued testing has confirmed that "cleaning up" certain file sets provides the following benefits:

- Improves time to open templates and objects.
- Improves time to check-in objects and templates.
- Deploying the InTouch app is faster.
- Restoring a Galaxy is faster.
- Backup was faster and smaller.

You can delete the files listed in the following sections. Doing so can shrink the Galaxy size dramatically. Although some of the directories can safely be deleted even while the Galaxy is deployed, it is strongly recommended that any deletions should be done when all the display nodes of a Galaxy have been un-deployed, (i.e. during an upgrade). When all the display nodes of a Galaxy are un-deployed, none of these directories should exist, so they can all be deleted.

**Important:** Always make backups before you delete anything!

## Application Versions

- Wonderware Application Server 3.1 and later (including 2012 R2 [3.6])

## Assumptions

This *Tech Note* assumes you are familiar with the following:

- Galaxy operations such as Deploy, Un-deploy, Object Export, Backup/Restore, etc.
- SQL Server Database management operations.

**Note:** See [Tech Note 976 SQL Server Transaction Log Not Truncating](#) for detailed diagnostic information about the SQL Server

## Delete \$\$\$ExportTempFolders

The **\$\$\$ExportTempFolders** are created when you do an Object Export from the Galaxy. These files are normally automatically deleted on completion. If the export fails or something goes wrong during the export, these directories are not deleted and remain there. All the \$\$\$ExportTempFolders directories can be deleted.

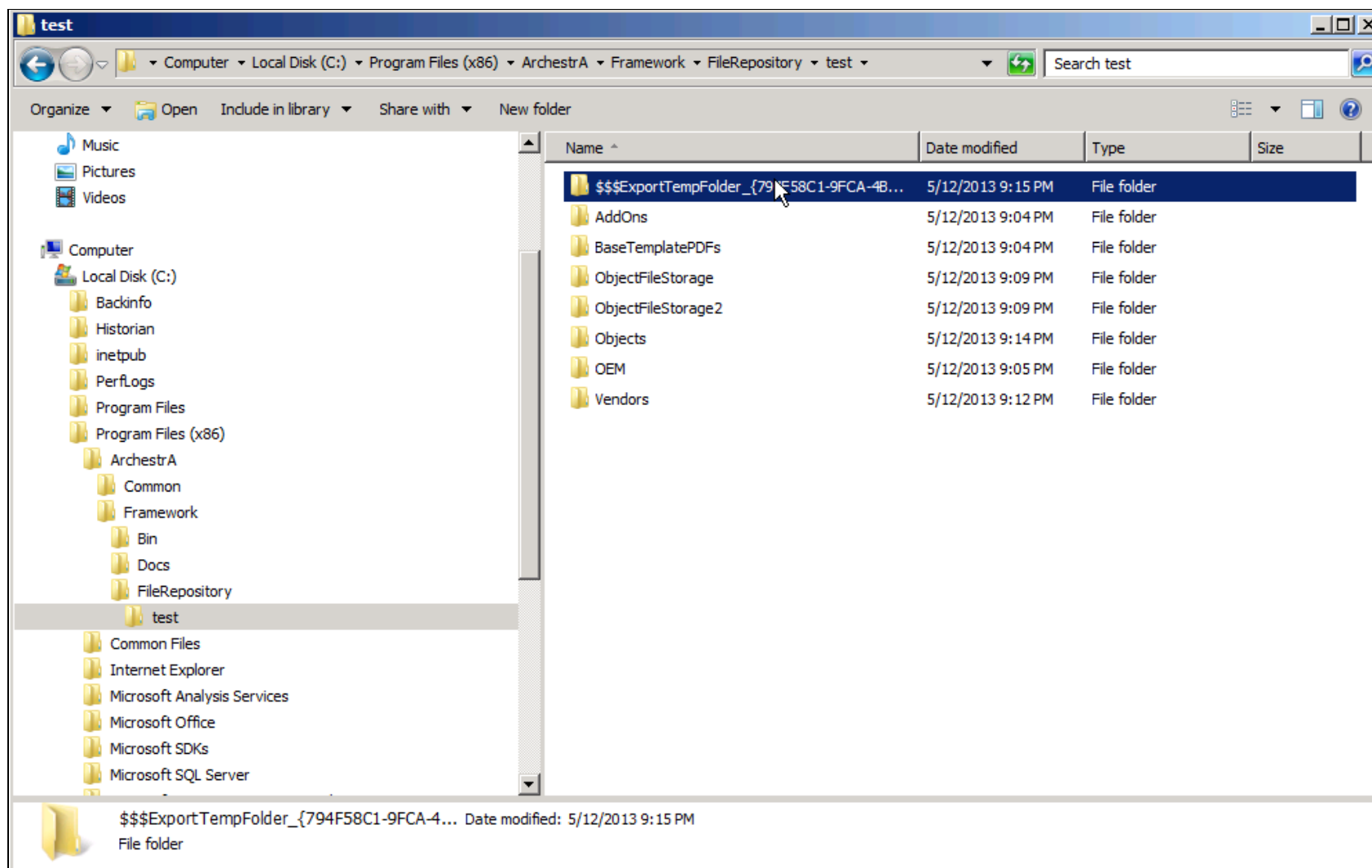


FIGURE 1: \$\$\$EXPORTTEMPFOLDERS

## Delete BAK Directories

The BAK directories are created when you make a backup of an InTouch Application when opening the application from within the IDE using WindowMaker. The default directory is in the application directory itself, so it permanently doubles the size of the application.

When a backup is required, specify an alternate directory and **do not** use the default. You can delete all the BAK directories.

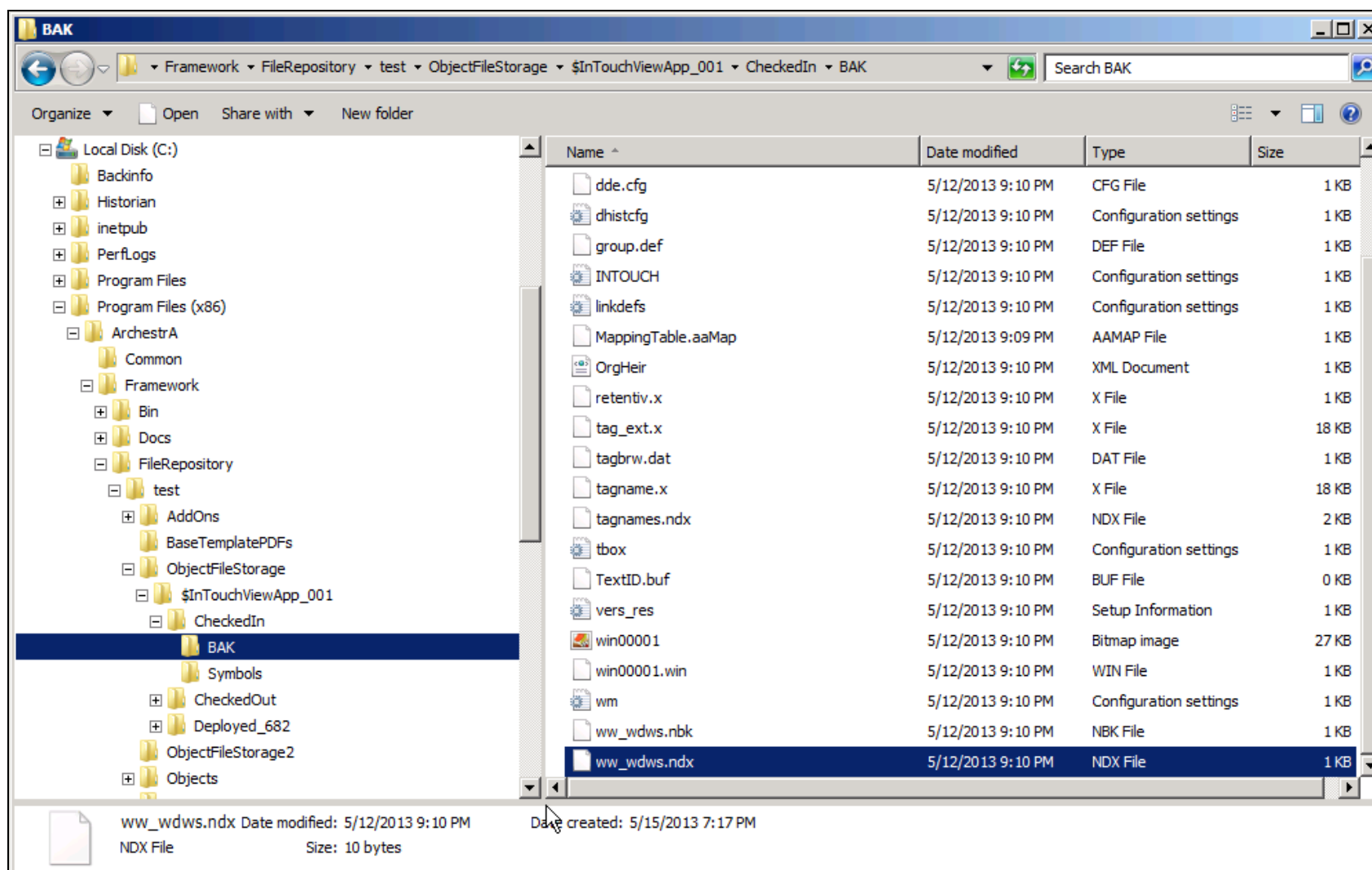


FIGURE 2: BAK DIRECTORY CONTENTS

## Delete Deployed Directories

**Important!** These files should only be deleted after all the display nodes of the Galaxy are un-deployed.

The Deployed directories are sometimes created when you deploy an InTouch application, but are not always deleted again when the

application is un-deployed. These files can build up over time.

- Un-deploy all the display nodes, and then delete all the directories left over. Afterwards deploy all the InTouch Applications again.

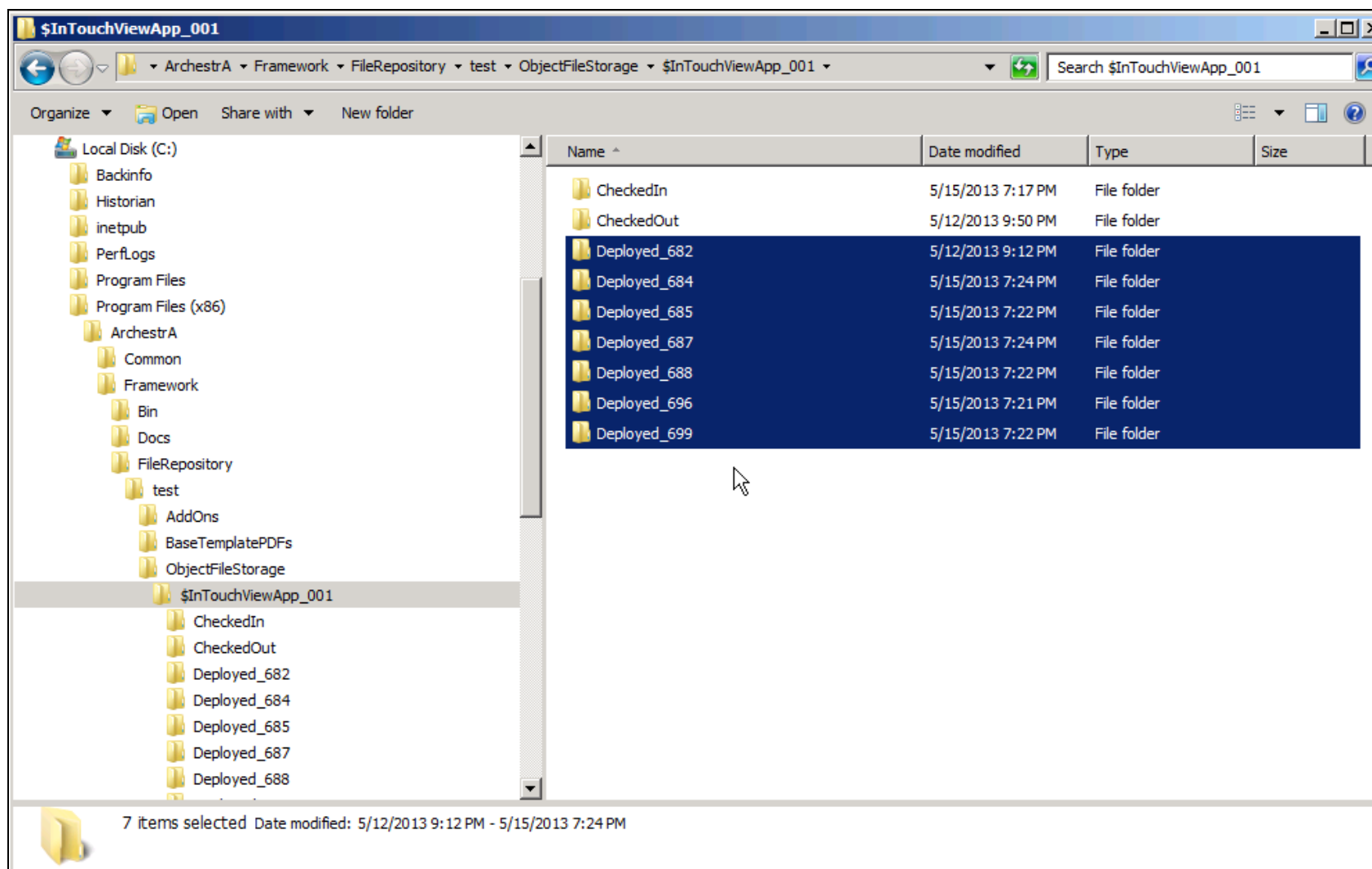


FIGURE 3: DEPLOYED DIRECTORIES

## Delete InTouch Application Directories

**Important!** These should only be deleted when all the display nodes of the Galaxy have been un-deployed.

For each deployed InTouch Application, a copy is created in this directory.

Various scenarios can result in these directories building up over time (Backing up and restoring the Galaxy, running Platform remove on

a node that hasn't been un-deployed, etc).

- Un-deploy all the display nodes, and then delete all the directories that are left over. Afterwards deploy all the InTouch Applications again.

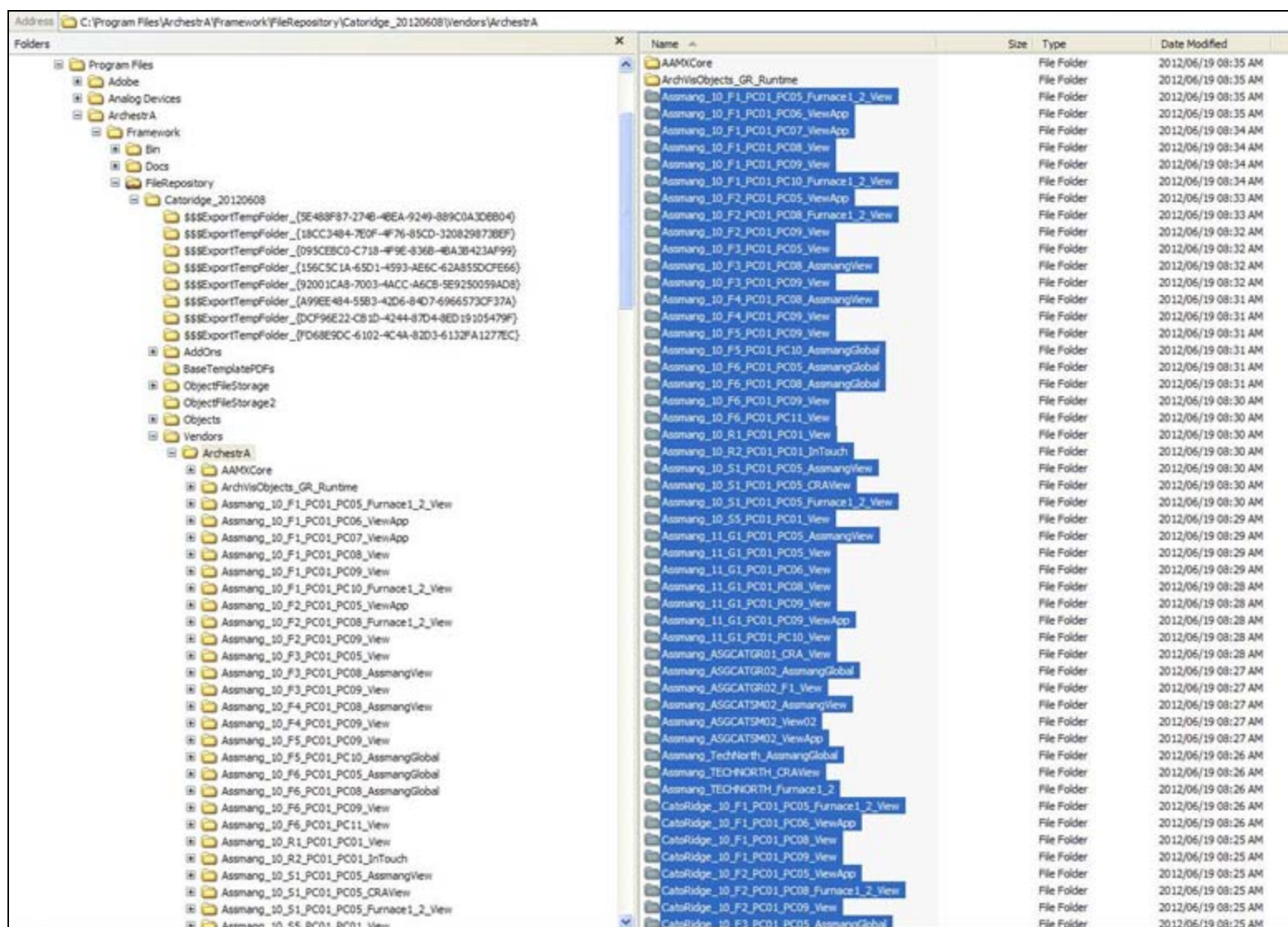


FIGURE 4: INTOUCH APPLICATION DIRECTORY COPIES

## Clear GOBJECT\_CHANGE\_LOG Table

The size of the table in the MSSQL Galaxy Database called **dbo.GOBJECT\_CHANGE\_LOG** can become very large over time when there is a large amount of development. This table can be cleaned up by executing the following script on the database:

USE Test

```
DELETE FROM gobject_change_log
WHERE change_date <= DATEADD(dd, -15, GetDate())
```

This query deletes all the entries in the change log more than 15 days old.

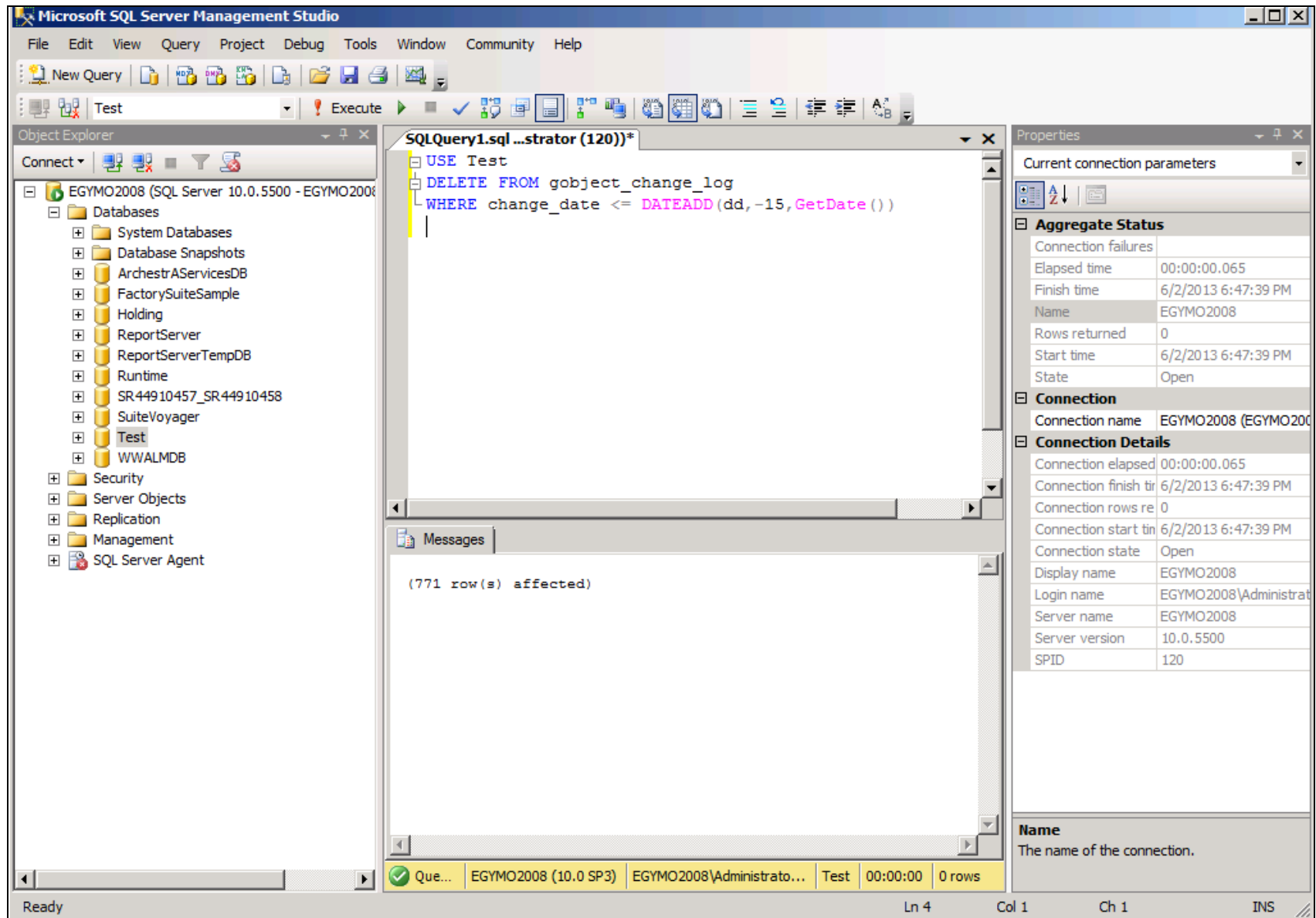


FIGURE 5: CLEAR THE CHANGE LOG

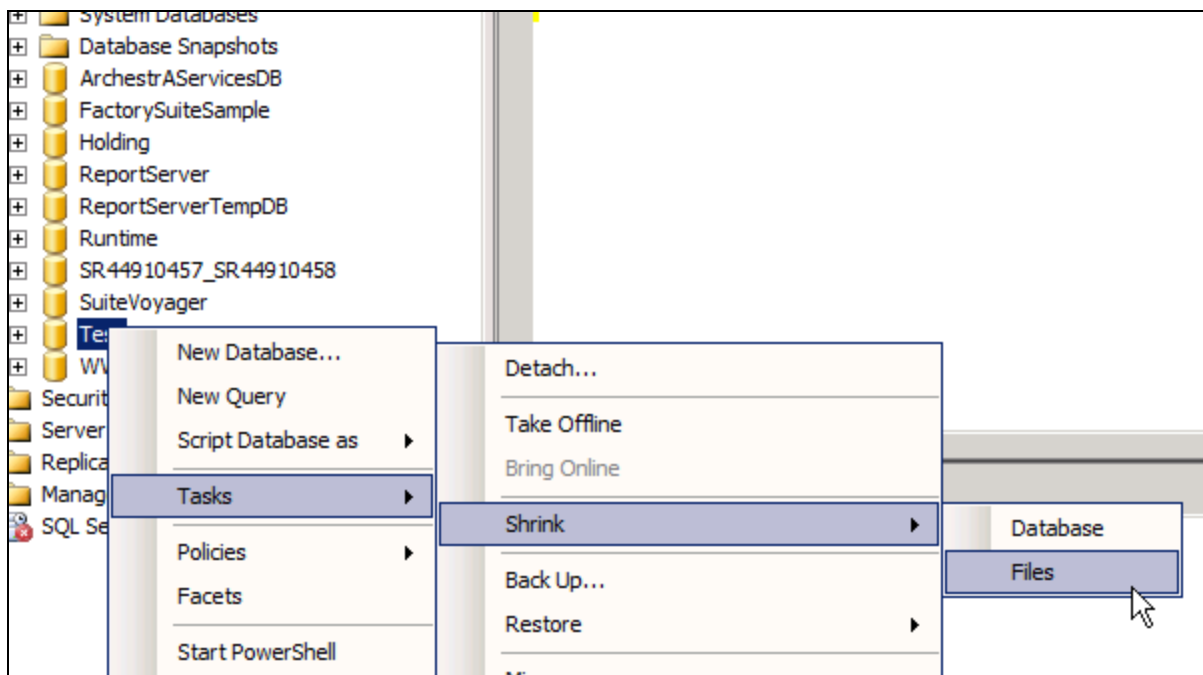
## Shrink Transaction Log Files

You can shrink the Database Log Files. In SQL Server Management Studio, select the Database which corresponds to the Galaxy Name



(e.g. **KH\_G**).

1. Right-click the Database item, then click **Tasks**.
2. Click Shrink, then Files (Figure 6 below).



**FIGURE 6: SHRINK DATABASE LOG FILES**

3. In the **File type** list, click **Log**, then **OK**. Leave all the other defaults.

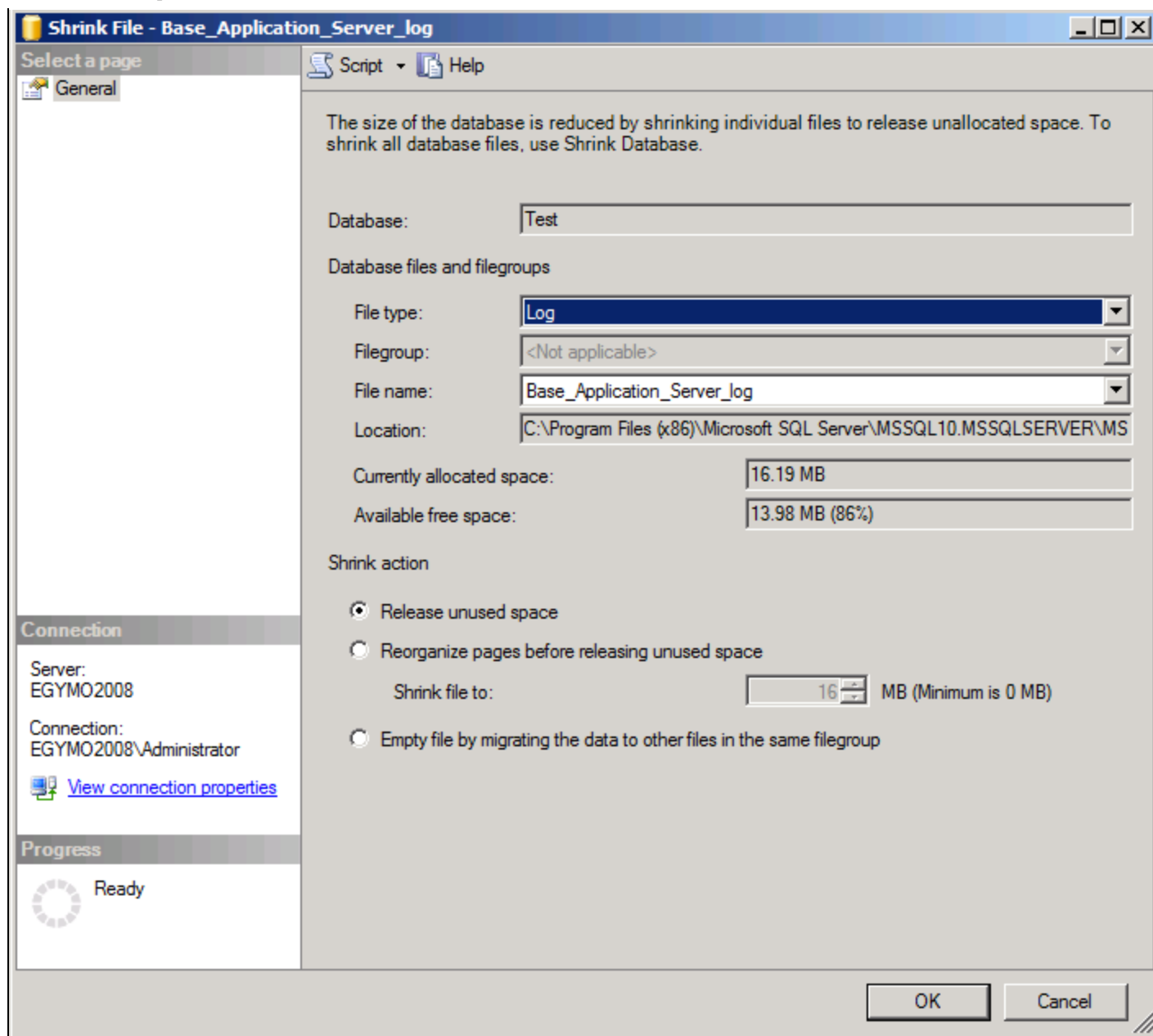


FIGURE 7: SHRINK FILE TYPE: LOG

- In this example, the original allocated space was **16.19** MB. After shrinking, it is reduced to **0.49** MB (Figure 8 below).



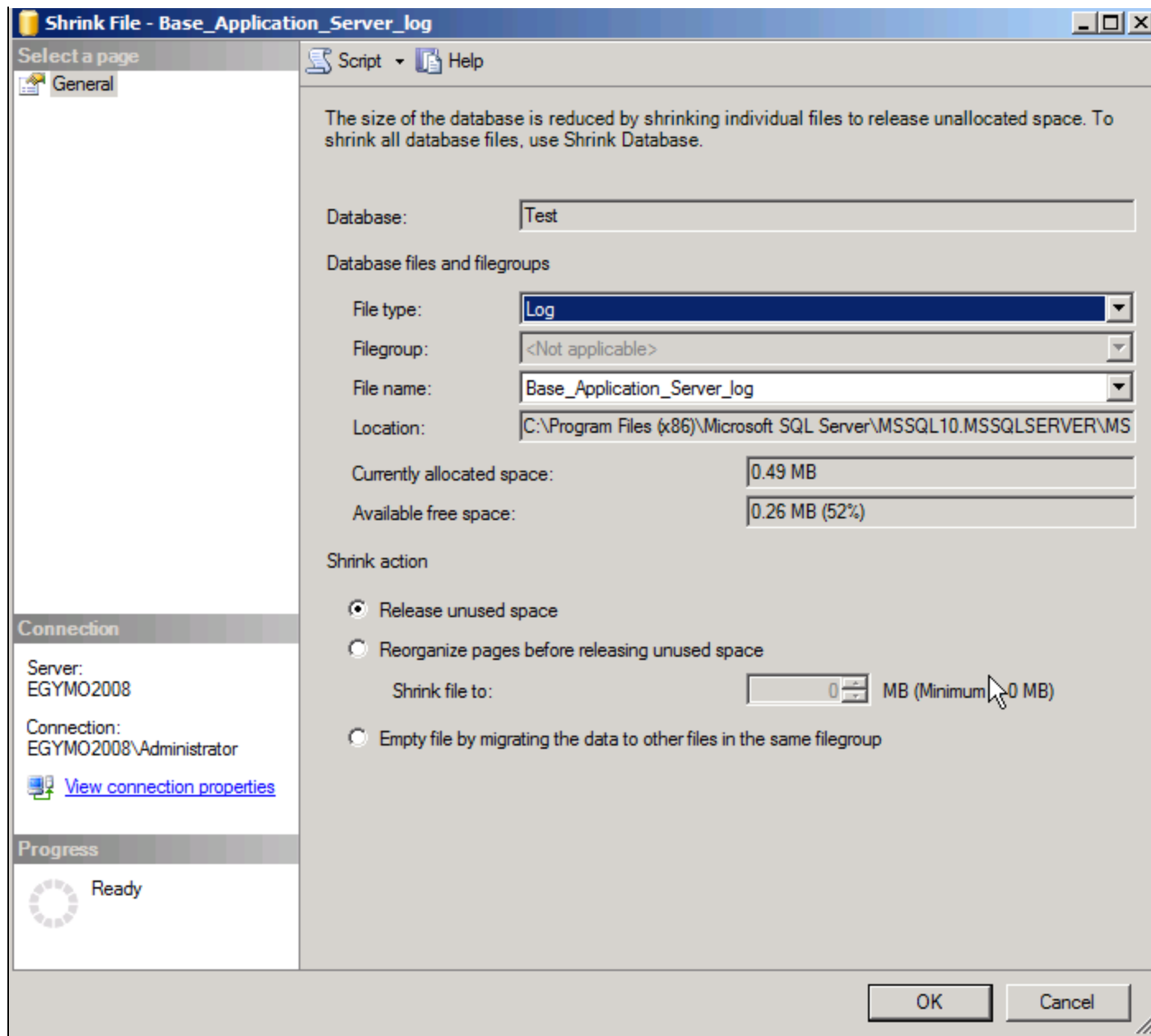


FIGURE 8: SHRINK FILE ALLOCATED SPACE

## Truncate the Log

You can truncate log data by using the following script:

```
DECLARE @DBNAME nvarchar(max)
SET @DBNAME= 'Test' /* Set the correct galaxy name here*/
DECLARE @RTp nvarchar(max)
```

```

SELECT @RTp = recovery_model_desc
FROM sys.databases WHERE name = @DBNAME ;
IF (@RTp is not null)
BEGIN
DECLARE @LOGNAME varchar(100)
DECLARE @sql nvarchar(max)
SET @sql = 'ALTER DATABASE ' + @DBNAME + ' SET RECOVERY SIMPLE'
IF (upper(@RTp) <> 'SIMPLE')
EXEC (@sql)
IF (coalesce(object_id(N'tempdb..##tt'),0)>0)
DROP TABLE ##tt;
SET @sql = 'select name into ##tt from sys.database_files where type =0'
EXEC sp_executesql @sql
DECLARE users_cursor
CURSOR FOR SELECT name FROM ##tt
OPEN users_cursor
FETCH NEXT FROM users_cursor
INTO @LOGNAME
WHILE @@FETCH_STATUS = 0
BEGIN
--Print @LOGNAME
DBCC SHRINKFILE (@LOGNAME , 1)
FETCH NEXT FROM users_cursor --have to fetch again within loop
INTO @LOGNAME
END
CLOSE users_cursor
DEALLOCATE users_cursor

SET @sql = 'DBCC SHRINKDATABASE (''' + @DBNAME + ''', TRUNCATEONLY)'
EXEC(@sql)
SET @sql = 'ALTER DATABASE ' + @DBNAME + ' SET RECOVERY FULL;'
IF (upper(@RecoveryModel) <> 'SIMPLE')
EXEC(@sql)
END

```

## Clean Up Unused Galaxy Objects

Execute the **internal\_delete\_unused\_packages** stored procedure to get rid of unused Galaxy Objects:

```

DECLARE      @return_value int,
             @all_finished int

EXEC @return_value = [dbo].[internal_delete_unused_packages]
    @all_finished = @all_finished OUTPUT

SELECT      @all_finished as N'@all_finished'

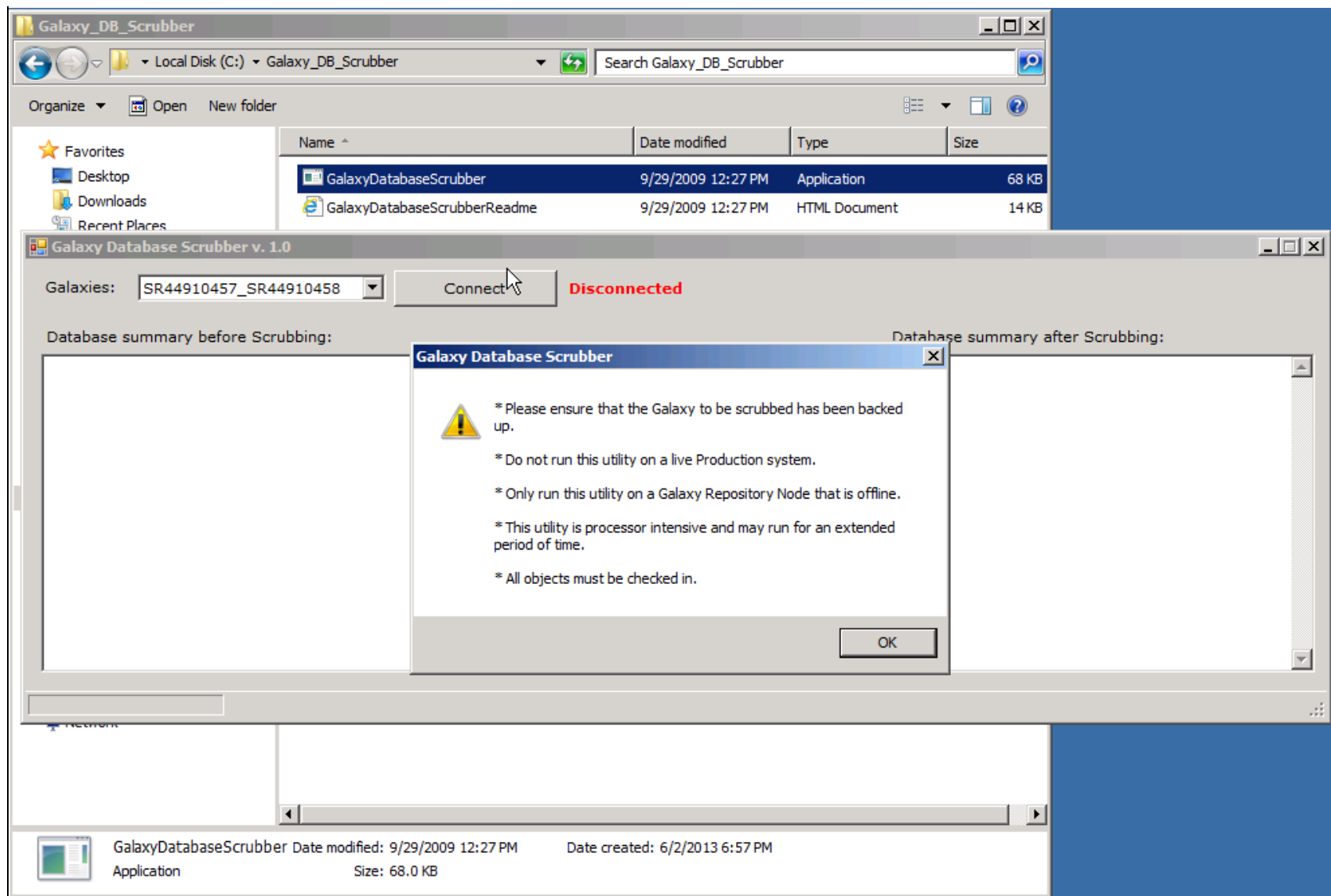
SELECT      'Return Value' = @return_value

```

## Run Galaxy Database Scrubber on Database

The Wonderware Galaxy Database Scrubber utility should only be run on Galaxies that have been completely un-deployed and backed up. This scrubber utility will delete all Unused Packages in the database and clean the Transaction Log files.

1. Double-click **GalaxyDatabaseScrubber.exe** to start the scrubber. Please take note of the warning pop-up:



**FIGURE 9: GALAXY SCRUBBER WARNING MESSAGE AT STARTUP**

2. Select the Galaxy that needs to be scrubbed and click **Connect**.
3. Click the **Scrub Database** button and wait until it is finished.

Wonderware South Africa and M. Salah

*Tech Notes* are published occasionally by Wonderware Technical Support. Publisher: Invensys Systems, Inc., 26561 Rancho Parkway South, Lake Forest, CA 92630. There is also technical information on our software products at [Wonderware Technical Support](#).

For technical support questions, send an e-mail to [wwwsupport@invensys.com](mailto:wwwsupport@invensys.com).

 [Back to top](#)

©2013 Invensys Systems, Inc. All rights reserved. No part of the material protected by this copyright may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, broadcasting, or by any information storage and retrieval system, without permission in writing from Invensys Systems, Inc.

[Terms of Use](#).