

# Taskmaster Client Service

---

## Roaming Setup and Operation

Datacap’s Taskmaster Client Service is a workflow utility that can continuously process *unattended* tasks without operator oversight. The *Guide to Taskmaster Client Service* describes the steps you take to install, test and run this service: the PDF version of this text is in the **Taskmaster (Additional)** folder of your **Datacap** directory’s **Docs** folder.

This document describes the **Roaming** feature that gives Taskmaster Client Service the ability to process unattended tasks of multiple applications, possibly via multiple Taskmaster Servers.

- ✓ **Important!** The level of the discussions and illustrations in this text assumes that you are familiar with Taskmaster Client Service. For your convenience, this documentation occasionally replaces “Taskmaster Client Service” with “TMSub”.

The examination of Roaming Setup and Operations includes the following topics:

<b>Overview</b> .....	<b>2</b>
<b>Setup</b> .....	<b>4</b>
<b>System Requirements</b> .....	<b>4</b>
Operating Systems .....	4
Files .....	4
<b>TMSub Roaming Configuration</b> .....	<b>5</b>
<b>tmroam.xml</b> .....	<b>6</b>
XML Tags .....	8
tmroam.xml - Additional Settings.....	9
tmroam.xml – Other Considerations.....	10
<b>TMSub Roaming – Operations</b> .....	<b>11</b>
<b>TMSub Roaming – Taskmaster Client Setup</b> .....	<b>13</b>
<b>Logs</b> .....	<b>15</b>

## Overview

Taskmaster Client Service's roaming capability permits the service to process designated Job/Task combinations in multiple *Taskmaster* applications – possibly accessing them via multiple Taskmaster Servers.

TMSub Roaming frees Taskmaster Client Service to move:

- between multiple applications of a single server;
  - from one server's applications to the applications of another server;
  - among Job/Task combinations within an application that do *not* require operator participation;
  - from individual Job/Task combinations to multiple Job/Task combinations covered by a single Job-Task shortcut icon.
- ✓ Taskmaster Client Service typically processes all batches that are available in one application before it roams to the next application (For details, see the **Operations** section on Page 11.)
- ✓ If an application is not available or responsive during processing, Taskmaster Client Service disregards that application until the Service is restarted.

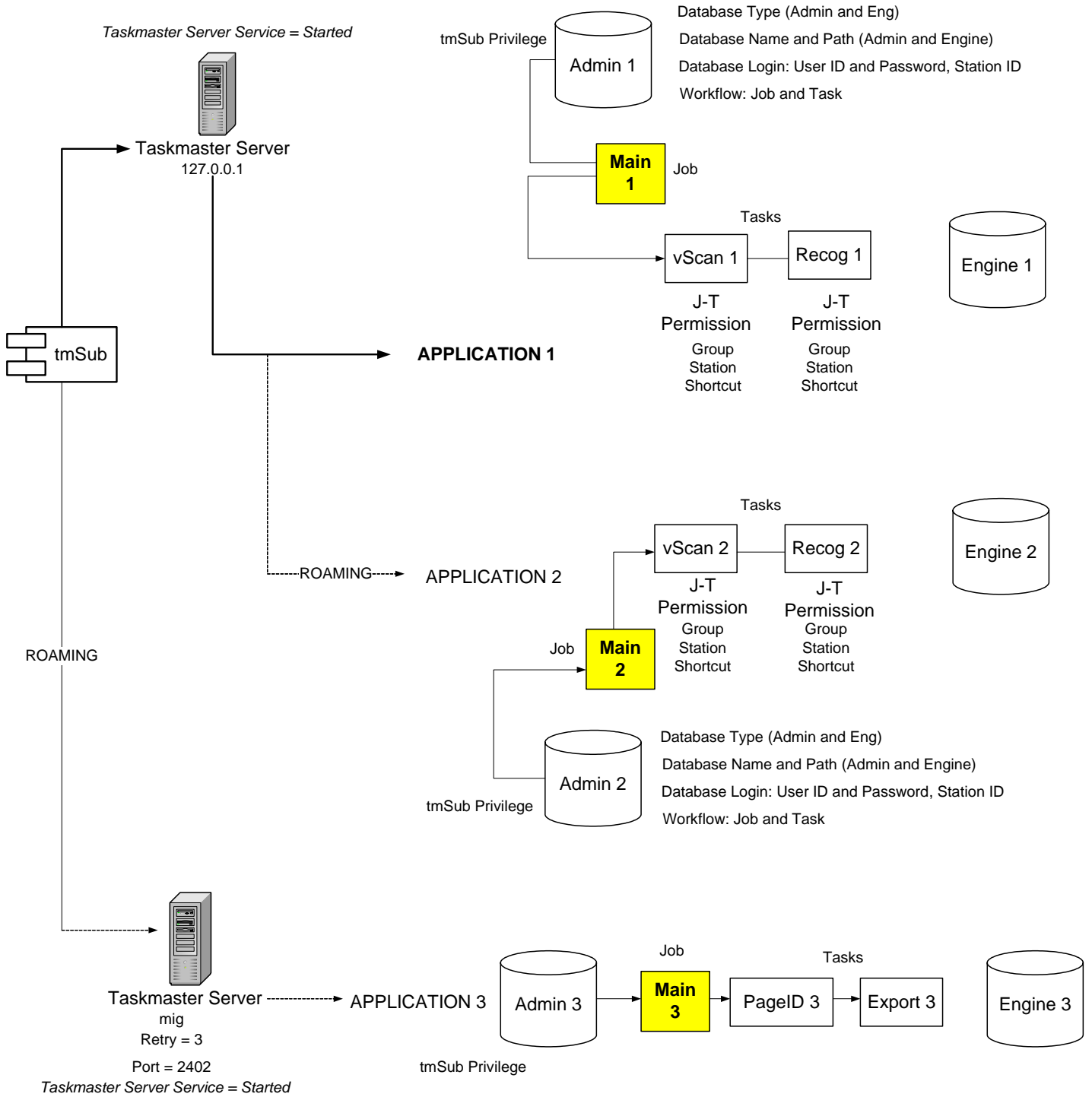
TMSub Roaming uses Windows Authentication as described in Datacap Tech Note #1319 ("LDAP Authentication"). Each *Taskmaster* application used with roaming must define a "user group" corresponding to a Windows group which the service Login Account belongs to. This Taskmaster Group's permissions limit the Job/Task combinations that Taskmaster Client Service can access.

Page 6 shows you how to define an XML file with roaming parameters. The chart on the next page depicts the possible scope of the service's roaming capabilities.

- The upper section illustrates how Taskmaster Client Service, in response to setup parameters, processes two Job/Task combinations of **Application 1**: the VScan task of the application's Main job; and the job's Recog task. Here, in its *regular* mode, the service works for one application only.
- **Roaming** first occurs when the service branches off to take care of two Job/Task combinations of Application 2.
- A second roaming episode occurs when Taskmaster Client Service is diverted to Job/Task combinations of Application 3, which is resident on a second Server ("mig").

The following sections focus on the nature and impact of roaming between the components depicted in this chart.

# Taskmaster Client Service: Roaming Setup and Operation



## Setup

Section II of the *Guide to Taskmaster Client Service* describes the installation of Taskmaster Client Service. Its topics include: the role of Taskmaster Server Service; settings of the .NET Framework; Account Permissions; and User and Station Permissions.

**Important!** Be sure to review these topics before you take steps to add the service's roaming capabilities.

## System Requirements

### Operating Systems

TMSub is supported on these operating systems:

- ◆ Windows 2000 Service Pack 4
- ◆ Windows XP Service Pack 2
- ◆ Windows Server 2003 or Windows Server 2003 Service Pack 1

TMSub requires MSXML 6.0 or MSXML 6.0 Service Pack 1.

- ✓ To install MSXML 6.0 Service Pack 1, you must have Microsoft Windows Installer 3.1 (MSI 3.1) or higher on your computer. MSXML 6 and MSI 3.1 are both available for downloading from [www.microsoft.com](http://www.microsoft.com).

### Files

Taskmaster Client Service employs three files which are in the `tmclient` folder of your host configuration's Datacap directory.

**TMSub.exe** is the executable file for the service. TMSub Roaming requires Version 7.2.0.20 or above. This is a Windows service and must be registered with this syntax: `"tmsub.exe-service"`.

**TMSubDock.exe** Version 7.1.2.21371 or above provides the User Interface for configuring TMSub.

**tmroam.xml** specifies the service's roaming parameters (Page 6)

In addition, the service requires updated versions of the following files (listed with their paths):

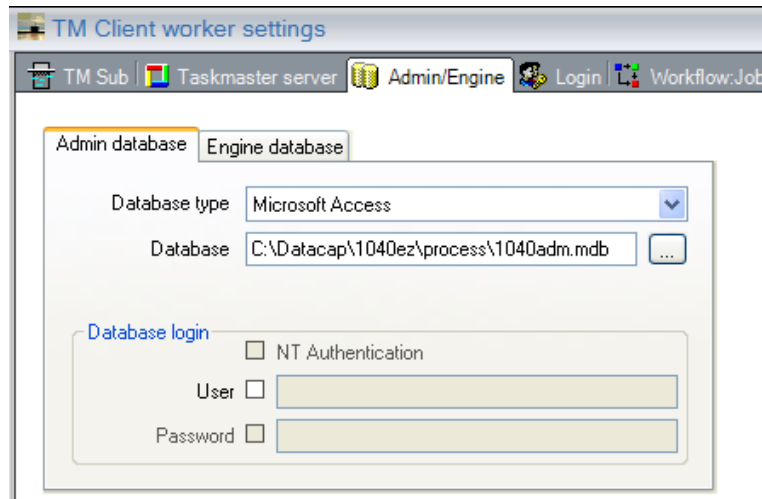
Datacap\dcshared\**TMrt.exe** (Version 7.2.0.1 or above)

Datacap\tmclient\**atm.dll** (Version 7.0.0.12 or above)

## TMSub Roaming Configuration

Taskmaster Client Service “roams” according to the hierarchy of instructions in the **tmroam.xml** file that you prepare and place in the **tmclient** folder of the host configuration’s **Datacap** directory.

**Very important!** When roaming, the service will override certain specifications in the tabs of the **TM Client Worker Settings** dialog. In the illustration below, a roaming service might ignore the **1040EZ** Admin database as it searches for an Admin database belonging to another application – or it might solicit information from yet other Admin databases.



TMClient Service Settings dialog – Admin/Engine tab

However, Taskmaster Client Service **cannot** operate in a regular or roaming mode unless you have completely set up the basic service, according to the procedures outlined in the *Guide to the Taskmaster Client Service*.

### tmroam.xml

Roaming is enabled when Taskmaster Client Service detects a properly formatted file named **tmroam.xml** in the **\Datacap\tmlclient** folder. The illustration below shows how a sample **tmroam.xml** configuration file defines roaming criteria for Taskmaster Client Service.

```
- <tm_roam>
- <server url="127.0.0.1">
- <db admin="PROVIDER=MSACCESS;DSN=C:\Datacap\1040ez\process\1040adm.mdb;"
  engine="PROVIDER=MSACCESS;DSN=C:\Datacap\1040ez\process\1040eng.mdb;">
- <job name="Main Job">
  <task name="Recognize" />
  </job>
  </db>
</server>
- <server url="mig" retry="3" port="2402">
- <db admin="PROVIDER=MSACCESS;DSN=C:\Datacap\mclaims\process\mclaimsadm.mdb;"
  engine="PROVIDER=MSACCESS;DSN=C:\Datacap\mclaims\process\mclaimseng.mdb;">
- <job name="HCFA Main">
  <task name="HC_Recog" />
  </job>
  <shortcut name="Background" />
  </db>
</server>
</tm_roam>
```

This file is a simple but powerful hierarchy:

File ID: tm\_roam

Server (1) url = 127.0.0.1

Application 1: Admin database (Connection String)

Application 1: Engine Database (Connection String)

Job: Main Job

Task: Recognize

Server (2): url = mig, retry = 3, port=2402

Application 2: Admin database (Connection String)

Application 2: Engine Database (Connection String)

Job: HCFA Main

Task: HC\_Recog

Job-Task Shortcut: Background

In this example, the service is directed to roam between two different servers; the application on each server is hosted by Access databases on the c: drive of that server. In the first instance, the service will run the Recognize task of the Main Job. In the second example, the service will process the HC\_Recog task of the HCFA Main job, and other unattended Job/Task combinations that can be launched by the Background shortcut.

The configuration below opens *two* applications in the mig server to roaming.



- ✓ **Please note:** The syntax of the `tmroam.xml` file must conform to XML requirements for a well-formed document, and contain the Roaming Tags described on the next page. If `tmroam.xml` is not syntactically correct, the Client Service will revert to non-roaming operation as determined by the configuration dialogs.

### XML Tags

The illustrations on the previous pages, and the table below highlight the tags that can be part of a **tmroam.xml** file. (The next page describes tags for optional settings.)

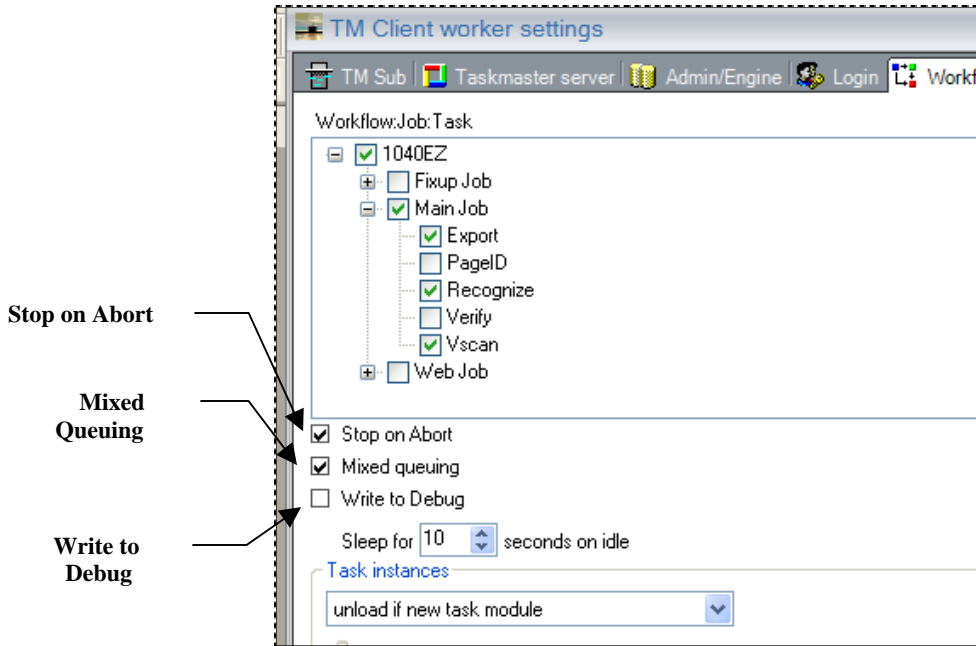
Tag	Level/Required?	Description
<tm_roam>	File <i>Required</i>	This tag opens the file and gives it a name.
<Server url =	Server <i>Required</i>	A server is a parent to applications – and this tag sets up information about one or more applications.  <b>Note:</b> Roaming cannot access any application without this specification.
<dbs	Database <i>Required</i>	This important tag announces the beginning of criteria for a single application.
Admin	Application <i>Required</i>	This is a Collection String that identifies an application's Admin database.
Engine>	Application <i>Required</i>	This second Collection String identifies the same application's Engine database.  <b>Note:</b> Together, the Admin and Engine parameters designate the application.
<job	Application <i>Optional</i>	When roaming, Taskmaster Client Service chooses among batches waiting for specific Job/Task combinations.  As a result, a file usually specifies both the job and task portions of the combination. Alternatively, the file can assign a single Job-Task shortcut icon to initiate one or more combinations.
<task	Job <i>Optional</i>	As the child of a job, a task is almost always linked in the file to a job. However, a Shortcut tag will eliminate the need for a job or task.
<shortcut	Application <i>Optional</i>	This tag directs roaming to process those Job/Task combinations launched by the shortcut.



**tmroam.xml - Additional Settings**

Taskmaster Client Service in its regular mode will respond to three settings in the *Workflow:Job:Task* tab of the **TM Client Worker Settings** dialog: *Stop on Abort*, *Mixed Queuing* and *Write to Debug*. (These settings are described in the *Guide to Taskmaster Server Service*.)

In its roaming mode, values in **tmroam.xml** will override those in the *Workflow:Job:Task* tab.



**TM Client Worker Settings dialog – Workflow:Job:Task tab**

The tags and possible values for the three setting are:

stop\_on\_abort = "y/yes/1/true" or "n/no/0/false"

q = "sequence" or "mix"

debug = "y/yes/1/true" or "n/no/0/false"

**Important!** There are no tags for other settings at the bottom of this tab; as a result, they cannot be part of **tmroam.xml**.

The example at the top of the following page illustrates the syntax of a **tmroam.xml** file with the additional settings.

Other  
settings

```
- <tm_roam>
- <server url="127.0.0.1">
  - <db admin="PROVIDER=MSACCESS;DSN=C:\Datacap\1040ez\process\1040adm.mdb;"
    engine="PROVIDER=MSACCESS;DSN=C:\Datacap\1040ez\process\1040eng.mdb;">
    - <job name="Main Job">
      <task name="Recognize" />
    </job>
  </db>
</server>
- <server url="mig" retry="3" port="2402">
  - <db q="sequence" stop_on_abort="y" debug="n"
    admin="PROVIDER=MSACCESS;DSN=C:\Datacap\mclaims\process\mclaimsadm.mdb;"
    engine="PROVIDER=MSACCESS;DSN=C:\Datacap\mclaims\process\mclaimseng.mdb;">
    - <job name="HCFA Main">
      <task name="HC_Recog" />
      <shortcut name="Background" />
    </job>
  </db>
</server>
</tm_roam>
```

### tmroam.xml – Other Considerations

All attributes and tags are in lower case.

Within each application, batches are selected using the queuing method configured in the Client Server Configuration dialogs, considering their Priority and in FIFO order by Start Time.

When *mixed queuing* is used, roaming “logic” processes the batches in each *Taskmaster* application in a depth-first manner. An application is defined by the Server URL and the Admin/Engine database pair. All the shortcuts and Job/Task combinations specified for the application are processed until empty (no batches remain pending); then, the Client Service will attempt to roam to the next application.

When *sequential queuing* is used, the roaming is breadth-first. In each application, Taskmaster Client Service attempts to run one batch for each designated Job/Task combination. Every batch-creation task will either run a pending batch or create a new batch. Normal tasks will run the highest-priority batch.

**Alert!** A Taskmaster Client Service that roams among applications with different Recognition engines can encounter difficulties. In the illustration above, if *1040EZ* uses an ICR/C engine to recognize data, and *MClaims* uses OCR/S, the service’s attempts to roam between the applications may not work correctly. If you run into problems, be sure to consult Datacap Support or your Datacap Implementation Specialist.

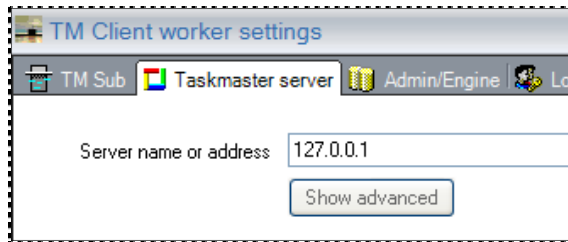
## TMSub Roaming – Operations

Section IV of the *Guide to Taskmaster Client Service* describes how the service processes unattended Job/Task combinations of a single application.

The service responds to specifications in the following tabs of the *TM Client Worker Settings* dialogs.

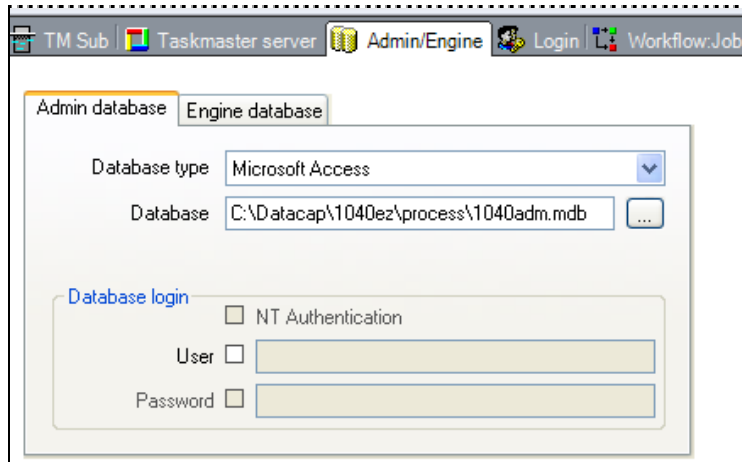
### Taskmaster Server Tab

The *Taskmaster Server* tab provides the server's name or address. For roaming, the server's name or address is obtained from **tmroam.xml**.



### Admin Database/Engine Database Tabs

These tabs indicate the names and paths of a single application's Admin and Engine databases. For roaming, **tmroam.xml** provides all database information, including details of the databases of other applications.



### Login Tab

Specifications in the *Login* tab give the service access to the application itself. Although Windows Authentication is a *Taskmaster* standard, you can select **Taskmaster Authentication** instead and enter a User ID and Password. **Note:** Roaming always uses Windows Authentication, despite the settings in the tab.

- ✓ In either case, you **must** enter a **Station ID**. In both its regular and roaming modes, Taskmaster Client Service requires this parameter because **tmroam.xml** cannot add a Station ID.

TM Client worker settings

TM Sub Taskmaster server Admin/Engine Login Workflow:Job:Task Logging

Taskmaster authentication

User ID

Password

Station ID

Windows authentication

Login

### Workflow:Job:Task Tab

This tab indicates which unattended Job/Task combinations will be processed by Taskmaster Client Service in its regular mode. For roaming, **tmroam.xml** designates alternative Job/Task combinations and shortcuts – and this tab is ignored.

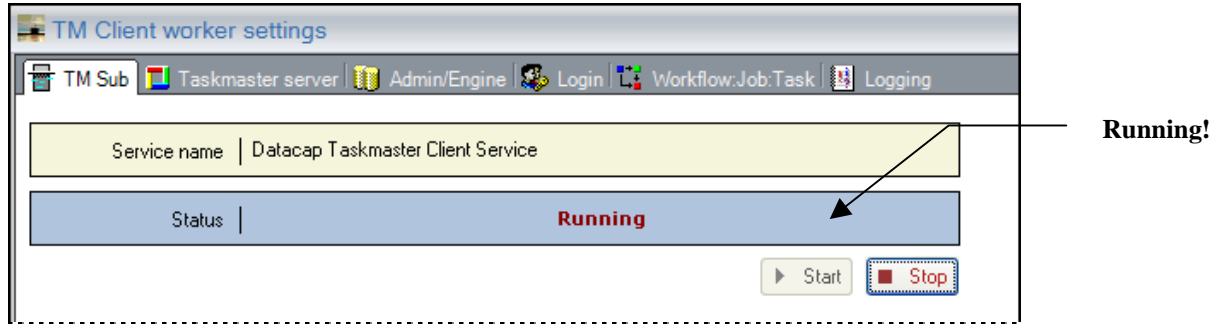
TM Client worker settings

TM Sub Taskmaster server Admin/Engine Login Workflow:Job:Task Logging

Workflow:Job:Task

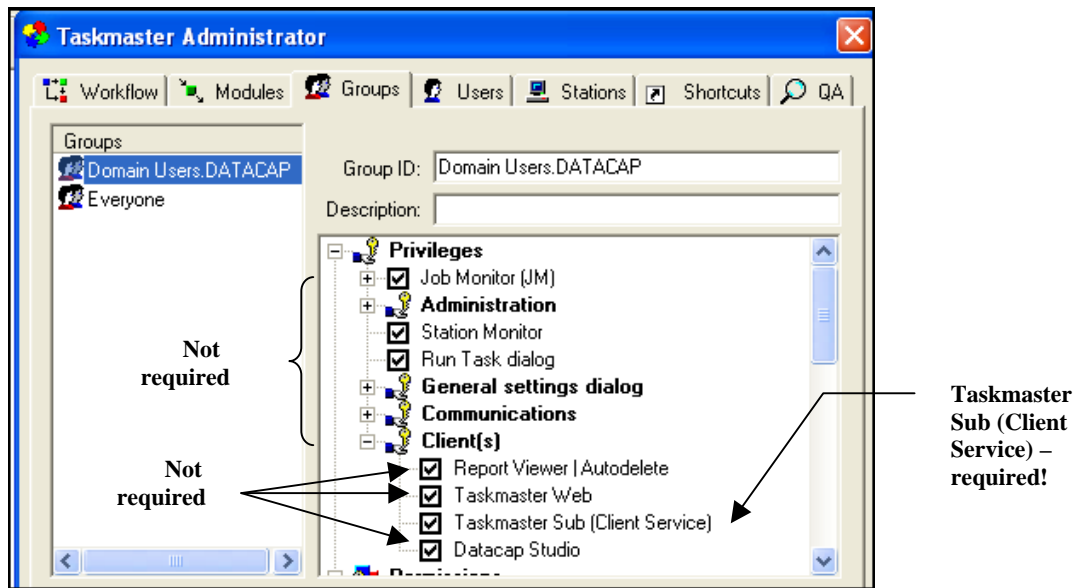
- 1040EZ
  - Fixup Job
  - Main Job
    - Export
    - PageID
    - Recognize
    - Verity
    - Vscan
  - Web Job

After these parameters are in place, Taskmaster Client Service begins working as soon as you click on the Start button in the *TM Client Worker Settings* dialog's *TM Sub* tab:



### TMSub Roaming – Taskmaster Client Setup

If Taskmaster Client Service is to successfully roam from one application to another, each application needs Taskmaster Group and Station permissions. These allow the roaming “user” to access the appropriate jobs and tasks.



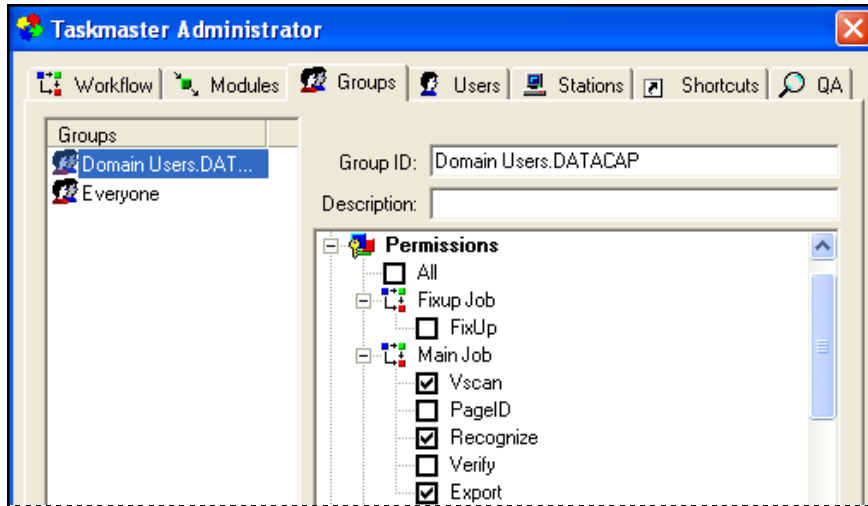
**Taskmaster Administrator – Groups tab**  
**Administrative Privileges**

The *Groups* tab of the application's **Taskmaster Administrator** must list an Active Directory Group that includes the TMClient Service Account that you are using.

In the example above, **Domain Users** is a group in the **DATA CAP** domain directory; the group includes the TMClient Service login account in the services control panel.

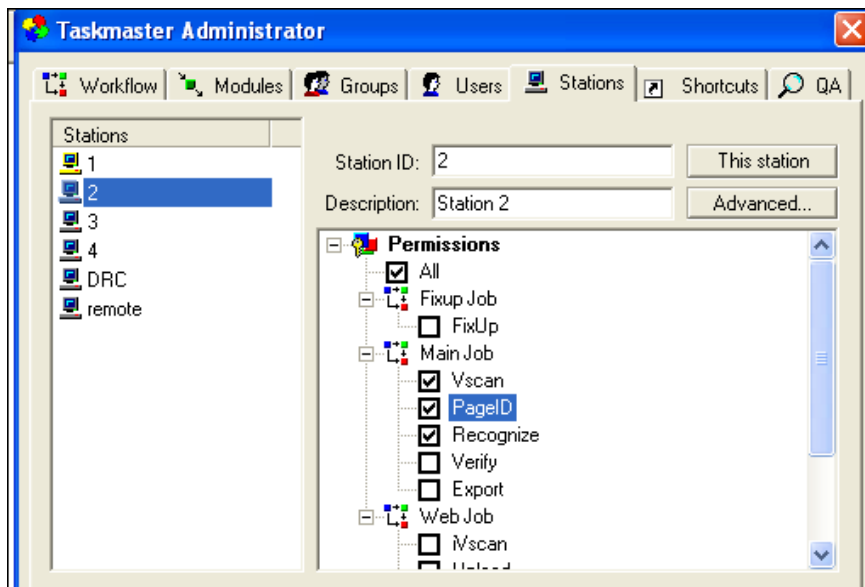
- *Taskmaster Sub (Client Service)* is one of the group's Administrative Privileges. This means that members of the group can launch and manage the service.

- Further down, the **Domain Users.Datacap** group has permission to run only three Job/Task Combinations: Main Job.VScan, Main Job.Recognize and Main Job.Export. **Important!** These are also the unattended Job/Task combinations you have selected for the Taskmaster Client Service in the *Workflow:Job:Task* tab of the *TM Client Worker Settings* dialog.



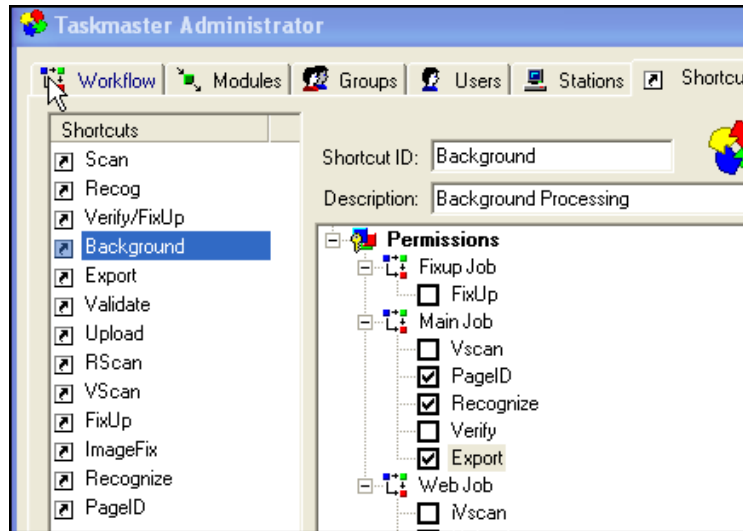
**Taskmaster Administrator – Groups tab**  
**Job-Task Permissions**

In the *Stations* tab, Station 2 can run three Job/Task combinations – but they are not exactly the same as those of the **Domain Users.Datacap** group.



**Taskmaster Administrator – Stations tab**  
**Job-Task Permissions**

In the *Taskmaster Administrator's* Shortcuts tab, the Background Job-Task shortcut also has permission to run three unattended Job/Task combinations.



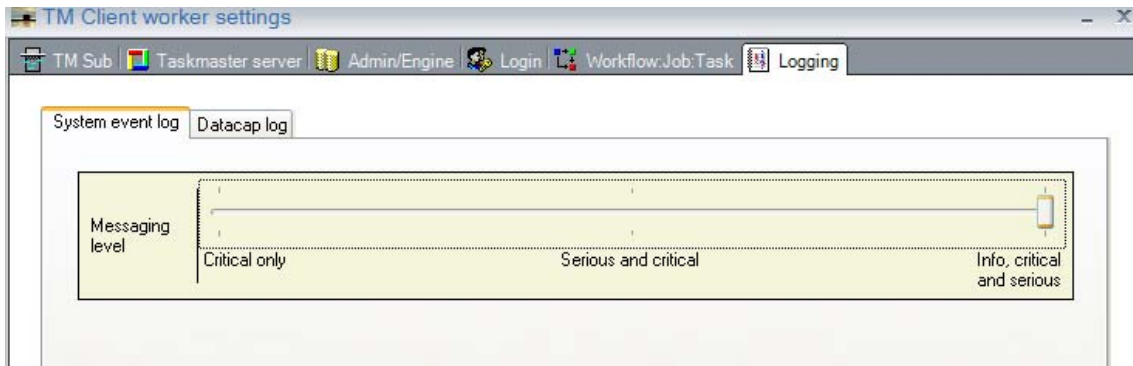
**Taskmaster Administrator – Stations tab  
Job-Task Permissions**

**Very important!** In its regular mode, Taskmaster Client Service cannot process a Job/Task combination without concurrent permissions from two *Taskmaster Administrator* components: the Windows Authentication Group and the Station.

If **tmroam.xml** cites a Job-Task **Shortcut** such as Background (see Page 7), the service can only process those of the shortcut's Job/Task combinations which can also be processed by the Windows Authentication Group and the Station.

## Logs

The level and amount of information about roaming in a Taskmaster Client Service log is determined by your entries in the *Logging* tabs of the *TM Client Worker Settings* dialog.

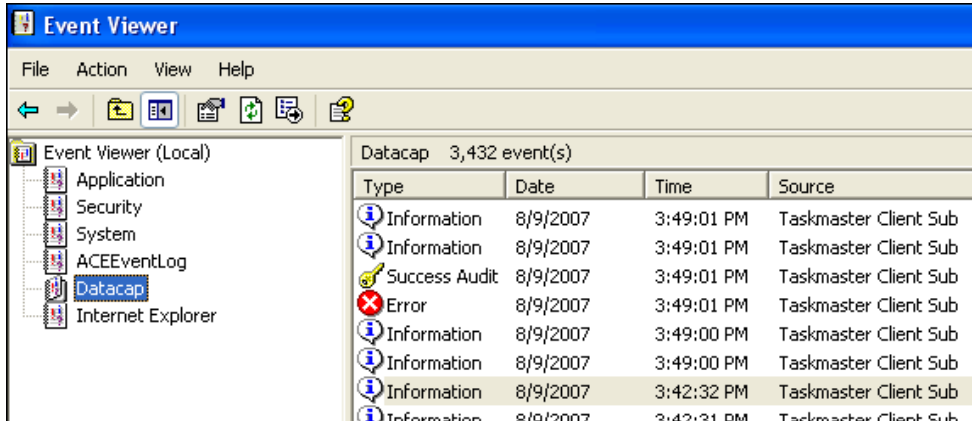


**Parameters for a System Event Log**

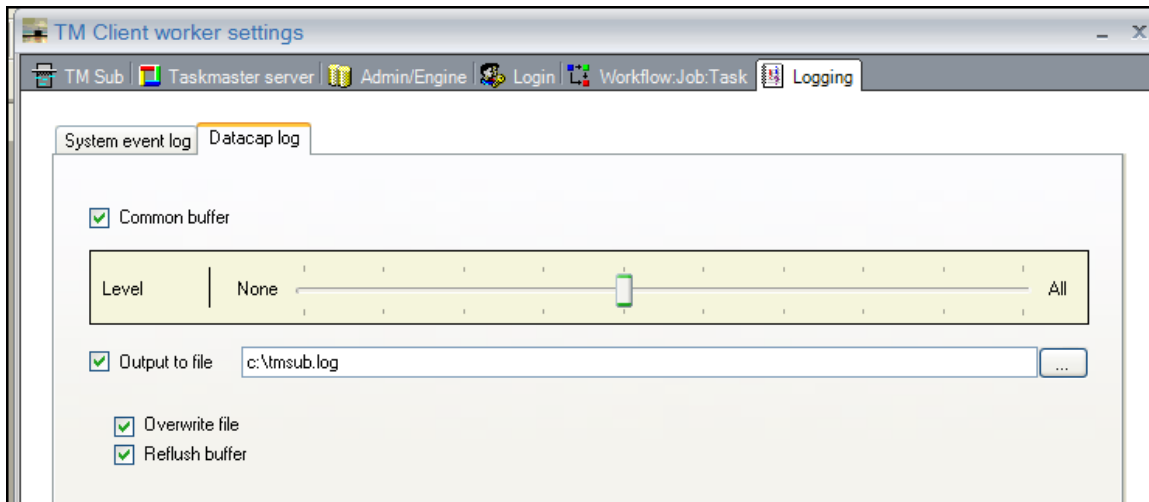
## TMSub Roaming – Operations

Settings in the *System Event Log* tab govern the kind of information you will find in an Event Log.

- ✓ To review this log, open the **Administrative Tools** of your **Control Panel**. Select the **Event Viewer**, and **Datacap**.



The *Datacap* log adds information to the file you indicate in this tab:



### Parameters for a Datacap

The file is easy to read through if you open it with a Text Editor such as *Notepad*.