

Chapter 3

Transactions

Transactions are generated in three components: Sampler, Transaction Pool, and Adder. Transactions are discrete entities that traverse the simulation network, and they can be used to represent physical and conceptual things such as a partially assembled refrigerator on an assembly line, a telephone call in a phone system, or an event such as a check on the size of a queue.

You can view the movement of transactions through the network by pressing the **Animate** button on the command panel. Bitmap images can be assigned to transactions in each of the three components in which they are generated. For example, the **Transaction Image** push button on the **Sampler** control panel opens a window from which you can choose an image to display on the transactions that originate at that Sampler. Figure 3.1 shows an example. Although Figure 3.1 shows the image SASHELP.ORIMG24.PHONE, any image in a SAS catalog can be used.

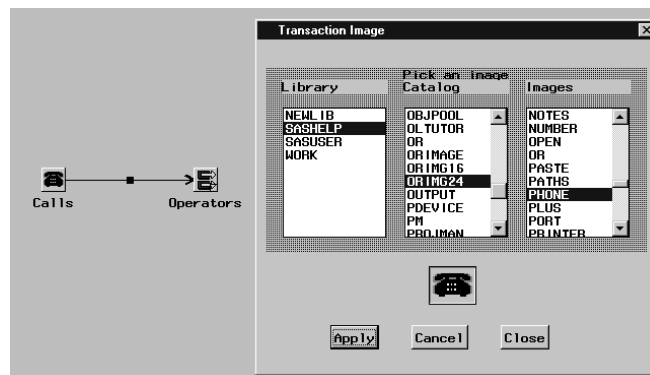


Figure 3.1. Setting an Image for a Sampler Transaction

When there are no references to transactions in the simulation the transaction is disposed of. This means that if a transaction does not reside in a queue, is busy in a server, is traversing an arc, or is being processed by another elementary component, it will be disposed of. You do not have to explicitly destroy them.

Attributes

You assign an attribute to a transaction in the Modifier component. There are three types of attributes: numeric, character, and unevaluated formulas. An attribute is attached to the transaction for the duration of the transaction, but it can be given another value at any time.

History

The transaction maintains a history of the time it has spent traversing the network. This history is saved in transaction attributes. For example, when a transaction enters

a queue, it records the time it entered the queue in an attribute that is named in the queue control panel (see Figure 2.5). When the transaction leaves the queue, it updates the attribute with the total time it spent in the queue. When a transaction enters a Bucket, if the **Collect Data** check box is set (see Figure 8.8), then any attribute in its history can be written to a SAS data set. You have to name the attribute for which you want to save data in the bucket control panel.

By default, the total age of the transaction is stored also a part of its history. This is the quantity that is by default collected in Buckets. You can access it with the attribute named “age.”

It is important to note that the transaction only spends simulation time in queues and servers. All the real time that is spent in other components and traversing arcs do not result in passage of simulation time.

Trigger Messages

The Trigger component includes an entry labeled “Transaction” in the Components list. Figure 3.2 shows a Trigger control panel with this entry selected.

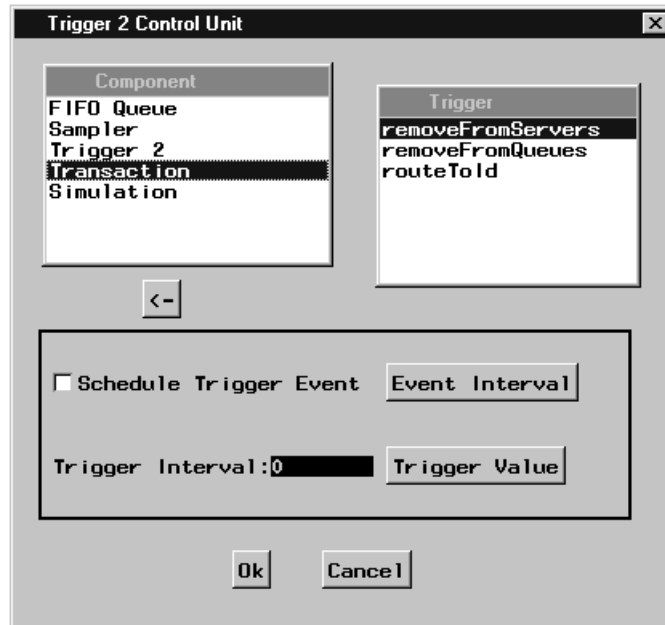


Figure 3.2. Transaction Trigger Messages

Notice the entries in the Trigger list box. These entries are possible behaviors to trigger when a transaction arrives at the Trigger component. In this example, because *Transaction* is selected, the target of the trigger is the transaction itself. The *removeFromServers* action causes the transaction to send that message to all servers in the simulation which in turn causes removal of that transaction from any servers in which it is receiving service. The other actions are described below.

removeFromServers removes the transaction from any server by sending the *removeIt* message to all servers.

removeFromQueues removes the transaction from any queue by sending the *balk* message to all queues.

routeToId routes the transaction to the component that has id equal to the value of the *routeToId* attribute attached to the transaction.

Note that when using the *routeToId* trigger you need to assign an attribute to the transaction with the name “routeToId” and the value the id of a component. You could assign that id as a numeric attribute when the transaction is in a modifier. This approach may not work if you save the model and then subsequently reload it because the id number of the component may change between the save the load. You can avoid this problem by assigning the *routeToId* attribute as the “id” of a model component.

These options provide you additional modeling flexibility. The two examples “Servers as Resources II” and “Special Routing” in Chapter 9, “Selected Examples,” illustrate two uses of the *routeToId* trigger.

Timing Transaction Arrivals

There are some issues concerning timing of transaction arrivals at a component. Consider model fragment A in Figure 3.3.

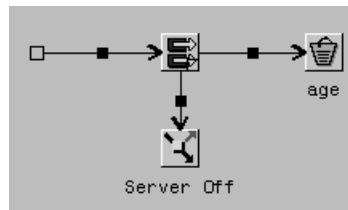


Figure 3.3. Model Fragment A

Although transactions leaving the multiple-server will arrive at the Trigger and the Bucket at the same simulation time, the logic and behavior of each of these components will be executed in sequence. There is no guarantee which will occur first.

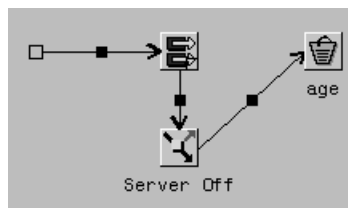


Figure 3.4. Model Fragment B

To guarantee that the Trigger “Server Off” executes before the transaction traverses to the Bucket, connect the components as in Figure 3.4.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/OR[®] User's Guide: QSIM Application, Version 8*, Cary, NC: SAS Institute Inc., 1999. 110 pp.

SAS/OR[®] User's Guide: QSIM Application, Version 8

Copyright © 1999 SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-488-8

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, by any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute, Inc.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999

SAS[®] and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. [®] indicates USA registration.

IBM[®], ACF/VTAM[®], AIX[®], APPN[®], MVS/ESA[®], OS/2[®], OS/390[®], VM/ESA[®], and VTAM[®] are registered trademarks or trademarks of International Business Machines Corporation. [®] indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

The Institute is a private company devoted to the support and further development of its software and related services.