Dynamic Transaction Routing

This section describes the dynamic transaction routing feature provided by Adabas System Coordinator:

- Benefits and Use

- Clustered Applications

Benefits and Use

Many TP systems can operate with a collection of jobs working together to provide a single TP service. Once this ability is available it becomes logical to load-balance client sessions across these jobs. The simplest implementation of this is to allocate new sessions to jobs on a "round-robin" basis. However, this does not cater to clients that are doing more work than others at varying times of the day.

The most effective way to load balance is to employ dynamic transaction routing. Algorithms are implemented into the TP system to move sessions from job to job according to availability and demand, thereby gaining the best throughput performance and service. This means that client sessions may be moved from one job to another at any time.

Using Adabas System Coordinator daemon mode makes it possible for Adabas client sessions to operate in a dynamic transaction routing environment. The client context for the options is made available automatically to all jobs in the service by the Adabas System Coordinator.

Clustered Applications

There is a standard service always running in the Adabas System Coordinator daemon called the Clustered Application Service (CAS). It is this service that provides support for dynamic transaction routing (daemon mode). In Adabas System Coordinator terms any TP service that is able to perform transaction routing is called a clustered application. The "cluster" refers to the fact that multiple jobs act together as members of one TP service.

Note:
Clustered applications are not supported in z/VM systems.

Clustered Applications in a Single Operating System Image

The following TP systems provide dynamic transaction routing across multiple jobs within a single operating system image:

- CICS/ESA or CICS Transaction Server for z/OS running with MRO (multi-region option)

- CICS for VSE/ESA or CICS Transaction Server for VSE/ESA running with MRO (multi-region option)

- IMS/ESA

- Universal Transaction Monitor (UTM) for BS2000

Configuration Example: CICS Cluster in a Single Operating System Image

The following graphic illustrates a CICS (MRO) cluster of two jobs running in a single system image. Terminal users (A001, A002) can be scheduled in any region and dynamic transaction routing can occur at any time.
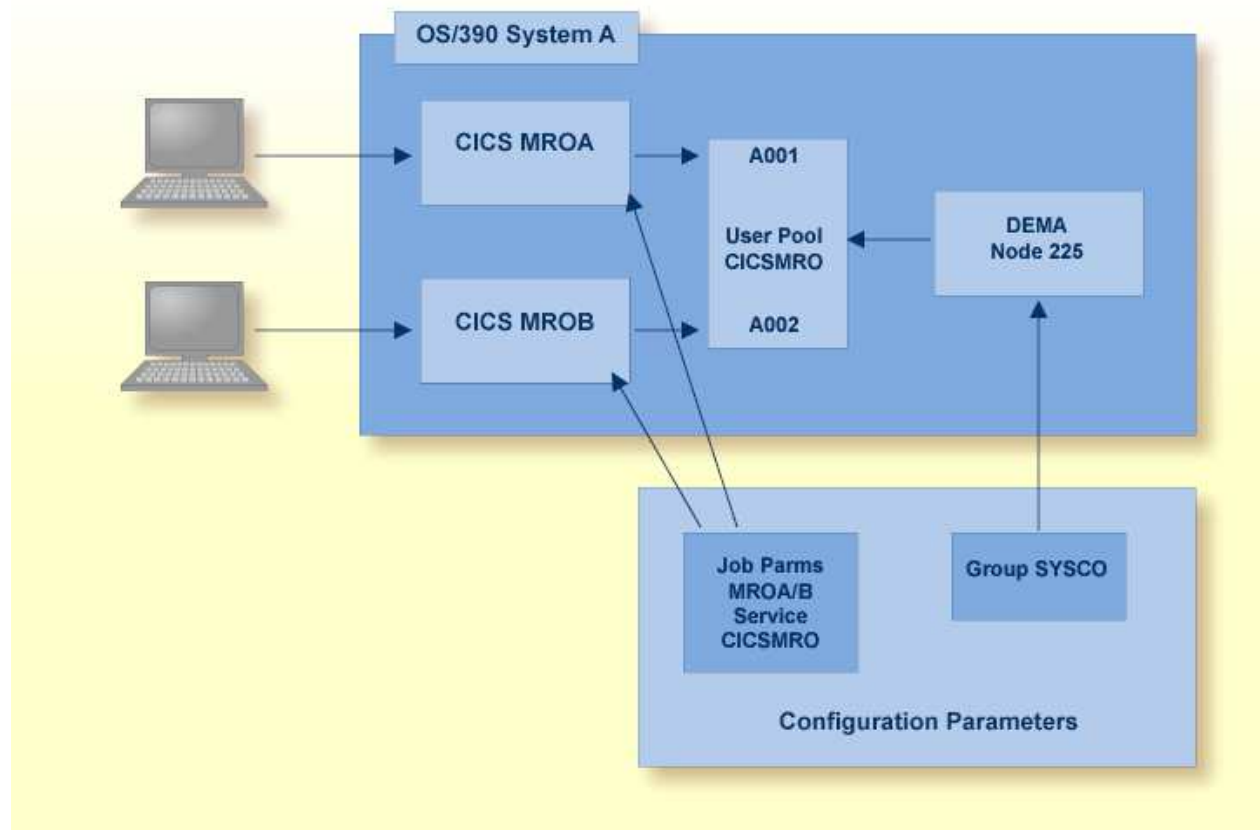
The Adabas option job parameters for the CICS regions CICSMROA and CICSMROB exist and reference the System Coordinator group SYSCO as responsible for enabling dynamic transaction routing. Both job parameters reference the service name CICSMRO, which enables dynamic transaction routing across the two regions.

The System Coordinator group has one member called DEMA defined to run using the node 225 in the Software AG network. The daemon DEMA must be started before any Adabas activity is generated in the CICS service.

Clustered Operating Systems

A clustered operating system connects multiple operating system images together to work as a single system image. The IBM parallel sysplex is an example of a clustered operating system. A clustered operating system improves throughput by providing more parallel capacity and more resilience for component failure.

A clustered operating system requires software that takes advantage of its cluster features. Adabas Cluster Services is used to access a single database resource from multiple Adabas nuclei operating on multiple operating system images coupled together in a sysplex cluster. Adabas Cluster Services balances loads across the nuclei to increase throughput and availability.

## Clustered Applications in a Clustered Operating System

There are TP services that support dynamic transaction routing across multiple operating system images such as CICS in a Parallel Sysplex. In these systems one Adabas System Coordinator daemon must run in each of the operating system images. The daemons work together as members of a peer group. If dynamic transaction routing is required throughout an IBM Parallel Sysplex, the Adabas System Coordinator daemons in the group must all share resources in the IBM Coupling Facility as described later in this section.

When a client session is routed from the domain (one system image) of one Adabas System Coordinator daemon to another, the Clustered Application Service (CAS) in each daemon negotiates the transfer of client context from one operating system image to the other. This occurs completely transparently to the running application.

Currently, transaction routing across operating system images is only supported in an IBM sysplex environment using IBM coupling facility features in a cache structure. An Adabas System Coordinator group must be defined and implemented in order to support this level of dynamic transaction routing.

Configuration Example: CICS Cluster in an IBM Parallel Sysplex

The following graphic illustrates a CICSplex cluster of two jobs running in an IBM parallel sysplex. Terminal users can be scheduled in any region and can migrate at any time.
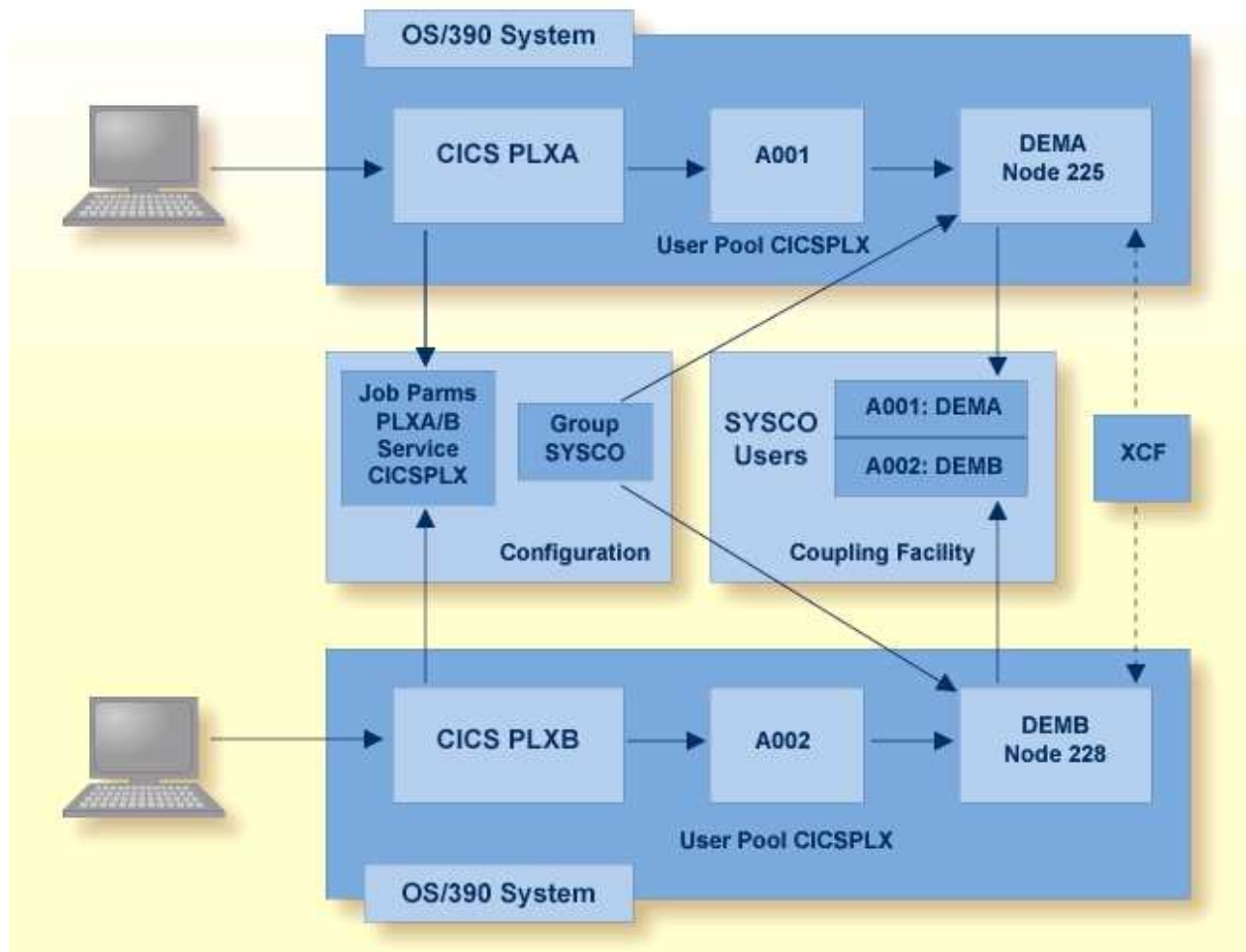
The Adabas option's job parameters for the CICS regions CICSPLXA and CICSPLXB exist and reference the System Coordinator group SYSCO as responsible for managing dynamic transaction routing. Both job parameters reference the service name CICSPLX, which becomes the name of the user memory pools on both systems and enables routing across the CICSplex.

The System Coordinator group has two members called DEMA and DEMB defined to run using the nodes 225 and 228, respectively. The daemons DEMA and DEMB must be started before any Adabas activity is generated in the CICSplex.

The System Coordinator group definition identifies the coupling facility cache structure to be used.

Adabas System Coordinator Group, Member and Service

Group

An Adabas System Coordinator daemon must be defined as a member of a System Coordinator Group with an eight-character group name (see Online Services, Maintain Daemon Groups). In a single operating system image there will normally be just one daemon member in the group. In a clustered operating system there will be one daemon member for each system image in the group. In an IBM Parallel Sysplex implementation the group name is also used as the XCF group name.

Jobs and TP services that are to be run in daemon mode identify the daemon group name to be used, rather than a specific daemon member name.

Member

The daemon member name is the job or task name by which the daemon service is known to the operating system.

Service

A single daemon member can provide DTR facilities to multiple TP services. Each discrete TP service (Production, Test, etc) must be identified with a unique service name within the daemon member. This provides isolation of one service from all others.

When the TP service starts up it registers its presence with the daemon. A shared memory pool is created for the service, for holding user context data. The service name is used for the pool name.

Identifying Clustered Applications to the System Coordinator

Support for dynamic transaction routing in a clustered application must be indicated in the job parameter for the Adabas option used. For example, to implement Adabas Fastpath, Adabas Vista, and Adabas Transaction Manager and support dynamic transaction routing in a CICS/MRO system, it is necessary to:

- create a job parameter using online administration services for one of the Adabas options,

- identify the job parameter as type "CICS cluster" rather than the standard CICS, and

- identify, as part of the job parameter, the Adabas System Coordinator group and service name to be used.

Adabas Fastpath, Adabas Vista, and Adabas Transaction Manager share the same job parameter in the same system configuration file. When the job type and system coordinator group are set in one Adabas option, they are used automatically by the others. Specifying a group and service name automatically identifies the job as a clustered application. When the job starts, it attempts to contact its local Adabas System Coordinator daemon to register its presence. All users that log on and use the application are also registered.

The Adabas System Coordinator daemon maintains user information in system shared memory such as ECSA in z/OS, not in local memory. The specific memory requirement differs depending on the activity level of the application and the Adabas options that are installed. To optimize the use of shared memory, which is a limited resource, Software AG recommends that jobs be defined as clustered applications only if dynamic transaction routing is to be supported in the job. It is still possible to administer groups from a job such as a TSO user that is not actually defined to an Adabas System Coordinator group.

Note:
Adabas Fastpath always uses an Adabas System Coordinator to manage its asynchronous buffer manager (ABM). When defining job parameters for Adabas Fastpath, however, it is only necessary to specify a System Coordinator group for clustered applications.