

Administration

- Administration
 - Configuration
 - Operation and Performance
 - Versioning
 - Node Error Program
 - Plug-in Service Routine
-

Administration

A design goal for all Adabas optional features as well as the Adabas System Coordinator is to provide the administrator with total control of the software operation from an online command center (administration tool). There is a deliberate focus on trying to avoid the need for control card inputs, JCL options, or parameter modules. In that it is not always possible to achieve 100% online operation, there are a few bootstrap configuration settings required but in essence the administrator can use the Adabas System Coordinator Online Services application for online administration and monitoring.

The Adabas System Coordinator Online Services tool (SYSCOR) is a Natural application which is used to administer the Adabas System Coordinator and the associated Clustered Application Service (CAS) by

- entering runtime controls for Adabas System Coordinator jobs and groups; and
- viewing active runtime information.

Configuration

- Configuration File
- Runtime Controls

Configuration File

Runtime controls are maintained in a configuration file that is held in an Adabas database. The configuration file is shared with Adabas Fastpath, Adabas Vista and Adabas Transaction Manager.

The configuration file contains:

- Runtime controls for client jobs and TP systems using Adabas System Coordinator and/or Adabas Fastpath, Vista, and Transaction Manager executing in client jobs and TP systems
- Runtime controls for the Adabas System Coordinator daemon

The configuration file is now a vital part of the runtime operation. As such it can become a single point of failure. Version 8.1 and above now allows you to nominate a primary and an alternate configuration file. Each session will attempt to use the primary and if it is unavailable the alternate will be used if it is nominated. Once a configuration file has been identified for a session that file will continue to be the primary file for that session until it becomes unavailable, and then the other file will be used. Consequently, over time different sessions may be using different files at the same time until you forcibly cause all sessions to switchover by making one or the other unavailable for a long period. If an alternate Configuration file approach is used then both files must be available at both Coordinator daemon startup and shutdown. This is necessary because the same recovery/restart information must be placed in both files so they do not get out of step.

Runtime Controls

The Adabas options and Adabas System Coordinator allow configuration by runtime controls through the administration tools. These can often be left to take their default values, and the defaults can be overridden for specific jobs when necessary. Consequently, the operation of specific jobs can be controlled remotely without having to gain access to the JCL. Furthermore, it is not necessary to install different various options in different libraries for use by jobs with different JCL. This provides for very flexible operational management.

Additionally there is increasing need for Adabas client sessions to operate differently within the same job. For example:

- Client ABC in CICSXYZ needs special tracing controls to be in use, all other clients do not
- Transaction D412 in CICSXYZ must be able to operate with a lower timeout limit than other transactions
- Stepname S0010 in job PRODA032 must be excluded from using the Adabas System Coordinator

This level of runtime control is becoming extremely important. For example, tracing options can be directed at a very few sessions rather than globally. This can mean overall memory consumption can be kept to a minimum while at the same time aggressively pursuing a problem investigating for only the sessions to be scrutinized.

Adabas System Coordinator allows these configuration controls to be prescribed in advance by adding optional override controls to the original base job level controls. It is possible to preconfigure overrides as follows:

- Batch job:
 - 1. Stepname
 - 2. LOGIN (for example, RACF LOGIN userid)
 - 3. Special API
- TSO, CMS, TIAM, etc
 - 1. Special API

- COM-PLETE, CICS, IMS, UTM
 - 1. Special API
 - 2. LOGIN
 - 3. Transaction code

For example, as a terminal operator moves from one transaction to another the runtime behaviors will alter dynamically according to what is prescribed in the configuration file. In addition to being able to pre-set the different configurations to be adopted at runtime it is also now possible to dynamically change the runtime controls for your "current" session. So, you may decide to switch tracing on or off, for example, regardless of what is prescribed in the configuration file.

Operation and Performance

- Operation
- Performance

Operation

The Adabas System Coordinator can be used in local or daemon mode by a job. The default is local mode. In local mode memory is allocated by the coordinator from within the job (process, region, partition, address space, etc.) in which it is running.

Alternatively, the administrator can configure a job to run in daemon mode. Obviously, this requires an Adabas System Coordinator daemon to be running in the same operating system image as the client job. In daemon mode the coordinator logic in the client job arranges for all context related memory to be managed by the daemon, enabling single- and multi-system dynamic transaction routing.

By using daemon mode for all jobs it is possible to use the administration tool to obtain feedback from all jobs in the system simultaneously. This is referred to as "single seat administration". This is one reason for using daemon mode. In local mode the feedback for a job can only be viewed from within that job since the allocated memory is only available within that job.

Daemon mode is also required for dynamic transaction routing as described in the next section.

Performance

The Adabas System Coordinator adds logic to the Adabas client, so it is understandable to assume that it brings with it some performance degradation. But this warrants further examination. For example, if the Adabas System Coordinator is used to house the Adabas Fastpath option, the chances are that performance is improved, not worsened. This obviously depends on the levels of optimization gained by the Adabas Fastpath functionality used and the relevance of the rules put in place.

The performance profile changes when Adabas Vista is used with the Adabas System Coordinator. The use of Adabas Vista functionality introduces an unavoidable overhead as the price of additional functionality. The Adabas System Coordinator helps to minimize this by providing highly tuned services, but there is nevertheless an overhead. If both Adabas Fastpath and Adabas Vista are used together with the Adabas System Coordinator, both benefit from a single context search so the combination of multiple options means that any overhead expected by an option on its own is lessened by the amount of shared

facilities. This is also true if Adabas Transaction Manager is added.

Each Adabas System Coordinator service is shared by all of the Adabas options. Therefore, in the future it may be decided to enhance various services to be more efficient. This will benefit multiple options at the same time. And, the reliability of each shared service helps to improve the reliability of all the options simultaneously. These are just some of the benefits of using a common technology framework.

The Adabas System Coordinator Online Services tool can be used to set job parameters and to obtain feedback from a running operation. For example, the list of Adabas clients being processed at the moment can be viewed to review certain statistics, such as the amounts of memory being used. These online facilities can prove very useful for locating and resolving performance problems.

Versioning

Introducing a new release of base Adabas collection, or the client-based add-on collection of products can be a big challenge in complex IT sites. Many sites will opt to update the base Adabas products separate from the client-based products to simplify the scope of the project and manage risk. More and more we have seen sites with stringent change-management that require you to perform implementation of new client-based releases in a very gradual, controlled fashion. This allows the switchover from one release to another to be managed job by job, client by client, database by database.

The Versioning Feature of Adabas System Coordinator enables this fine-grained ability to perform upgrades in a very stealthy, managed way that provides great benefits to your goals for continuous operation. By using the versioning feature you can:

- Run two releases of client-based products within an Adabas database
- Run two System Coordinator daemon versions in the same system-image
- Run client jobs that use different releases

Consequently, you are able to convert one client job at a time, step by step, until all clients are running the new release. At that point you decommission the daemon running for the old release and you decommission the old release within your databases.

Node Error Program

The node error program CORNEP is used by sites running CICS command-level applications in CICS/ESA, CICS Transaction Server for z/OS, CICS for VSE/ESA, or CICS Transaction Server for VSE/ESA.

CORNEP is not an essential component, but it does provide efficient memory reclamation for user sessions that terminate without releasing precious memory resources.

Important:

Use of CORNEP requires modification of your installation CICS DFHZNEP program. CORNEP must only be called from DFHZNEP.

Plug-in Service Routine

The Adabas error handling and message buffering facility helps implement 24*7 operations by analyzing and recovering from certain types of errors automatically with little or no manual intervention. It also generates additional information so that the error can be diagnosed. See the *Adabas DBA* documentation for more information.

To work within this feature, the Adabas System Coordinator delivers a plug-in service routine PINCOR, which is established automatically when the Adabas System Coordinator server component (ADAPOP) initializes at nucleus startup.

If a program interrupt occurs in the Adabas System Coordinator server component, control is passed to PINCOR, which formats and prints the main memory areas used by the component.

These diagnostics are written to the DDPRINT dataset with the following title:

```
COMMON RUNTIME - memory-area-name : SNAP BY SMGT
```

PINCOR then returns control to the error handling and message buffering facility so that Adabas can terminate abnormally.