

# FINACLE SUPPORT CONNECT

## Knowledge Series

July 2021 | Volume 31

Finacle Support brings you this fortnightly knowledge bulletin to augment your problem-solving capability. There is more to it. Every edition is put together with utmost diligence to ensure that best practices and known resolutions are shared. In this edition you will find the following articles:

- **Customization Options in Asset Classification or HSASCL Process**
- **Commonly used FI APIs for FAS Application**
- **Analysis of Verbose Garbage Collection Logs (VGC)**

So let's start reading!

### Customization Options in Asset Classification or HSASCL Process

**Product: Finacle Core Banking Version: 10.x onwards**

In Finacle Core Banking, Asset classification is performed through **HSASCL** menu to either upgrade or downgrade the entities based on the delinquency cycle. Some of the significant **COMT** parameters, **Env** variables, and associated script hooks are listed below, which can be utilized to optimize the **HSASCL** process.

List of **COMT** parameters and **Env** variables:

- **MAIN\_USER\_CLASS\_EXEMPT**: This is an env variable which is usually exported as the default main asset classification value set in HSCFM. This will be used to exempt entities

from **SASCL** process pertaining to the default asset classification provided there are no outstanding liabilities for those entities. This will improve the performance of HSASCL job, as unwanted entities are not processed by the job

- **EXEMPT\_SCHM\_TYPE**: This is an env variable which is usually exported as a scheme type or group of scheme types with **SPACE** as delimiter. The entities belonging to these **Scheme Types** will be exempted from **SASCL** process
- **SKIP\_COMPUTE\_CRAMT\_INTAMT**: If this set as **Y**, system will skip computation of
  - **Sum of Dr. Interest Amount** since the last **Asset Classification Date**
  - **Sum of Cr. Transactions** since the last **Asset Classification Date**
  - **Number of days** since the last Cr. Transaction

For banks, where this logic is not needed, removing or commenting (in **commonenv**) will improve performance of the batch

- **PRI\_SOL\_BASED\_ASSET\_CLASSIFICATION (COMT variable, 10.2.18 and above)**: If this is set as **Y**, then system will match the input SOL ID/SET ID with the Primary sol id of a customer and process such entities in the HSASCL process

#### List of Product script hooks:

- **sys\_asset\_classfn.scr**: Can be used to override (upgrade or downgrade) the system-derived Asset classification values. This script needs to be specified in **HSRGPM** parameter set up for the required scheme codes
- **getMaxDPDCount.scr**: This script can be used to modify the system-derived **DPD** counter values (by assigning the new **DPD** value in the variable **BANCS.OUTPUT.NewDPD**), which will further influence the corresponding asset classification by **HSASCL** process based on this **DPD** counter value
- **getSumDpdValues.scr**: Can be used to reset the **DPD** value with the sum of current **DPD** and **DPD** before reschedule for the accounts. E.g, the system adds **DPD values before reschedule** and **DPD in current schedule**, during the **HSASCL** process after delinquency rescheduling
- **CheckIfPercolationReqdForAsset.scr**: Can be used to exclude accounts from the percolation process for any bank-specific requirements. The **BANCS.OUTPUT.successOrFailure** variable can be set accordingly
- **CustomAccList.scr (10.2.18 and above)**: Can be used for executing **HSASCL** process for specific list of accounts. The **DB** query for selection of **ACID** values can be written in this script

## Commonly used FI APIs for FAS Application

**Product:** *Finacle Alerts Solution Version: 10.4.02, 10.5 onwards*

Finacle Alert Solution maintains registration details and alert subscriptions for the customer. These details can be added to **FAS** using **FAS Admin Application** and/or batch program.

Apart from the admin application and batch, there are FI APIs available for **FAS** which can be integrated with host systems like Online Banking, CRM etc., for real time communication between **FAS** and Host Systems.



Given below is the list of APIs which are most commonly used and their functionalities.

PURPOSE	FI API Name	ACTION
Customer Registration	EASCustAdd	Add External customer details to FAS
	EASCustMod	Modify existing external customer details to FAS
	EASCustDel	Delete an existing external customer
	EASCustFetch	Fetch details of an existing customer
Alert Subscription	CustAlertSubscribe	Subscribe customer to an alert
	CustAlertSubscribeDel	Delete exiting customer subscription for an alert
	CustAlertSubscripMod	Update exiting customer subscription for an alert
	CustAlertSubscripFetch	Fetch subscription details for specific customer
Publishing Event	AlertEventPublish	Makes an entry into <b>REPT</b> table of FAS for user details and alert details verification
Alert Delivery	AlertMsgDeliver	Makes an entry into <b>PUDT</b> table of FAS with the Alert details which needs to be delivered go the end user
Status of Alert	AlertMsgStatusInq	Inquire the status of delivered alert messages

## Analysis of Verbose Garbage Collection Logs (VGC)

**Product:** *Webserver Application Server (WAS)*

For the deployed application, the requirement of Java heap size may increase for some reason. JVM will not allocate memory beyond max heap size. This can lead to frequent garbage collections and out of memory error can occur. In order to monitor JVM heap for the deployed application, JVM verbose garbage collection logging can be enabled for the Java application server.

**Verbose Garbage Collection Logs (VGC)** Analysis includes details of the entire GC process like duration, objects promoted, OOMs, memory leaks and the resources it uses, which in turn helps in improving the performance of JVM applications.

**VGC** logs can be enabled by checking the **Verbose Garbage Collection check box** in the admin console at the below paths based on the **UNIX** flavor.

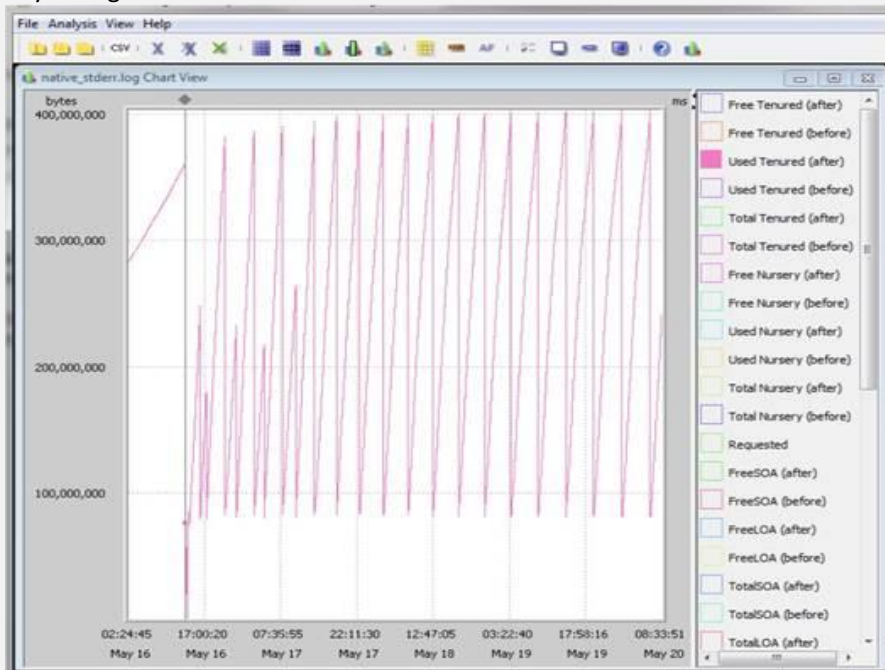
### WAS on AIX/Linux:

- Login to **WAS console > Servers > Server Types > WebSphere application servers > {server-name} > Java and Process Management > Process Definition > Java Virtual Machine > Verbose Garbage Collection** check box. By default, GC logs will be written to **native\_stderr.log** file
- To redirect **GC** logs to a custom file and to enable log rotation, add the below argument under **Generic JVM** arguments. While adding the argument, replace <PATH> with actual WAS logs path.  
**-Xverbosegclog:<PATH>/verbosegc.%Y%m%d.%H%M%S.%pid.txt,5,20000**

### WAS on HP-UX/Solaris:

- Login to the console and enable the **VGC** checkbox
- Following JVM arguments to be added in the admin console, under **Generic JVM arguments > -XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+PrintHeapAtGC**

For any clarifications, please reach out to your Support Manager.  
The image below shows a healthy GC log.



Hope you like this edition. Is there anything that you'd like to see in the forthcoming series? We'd love to hear from you!

Write to us at [finaclesupport@edgeverve.com](mailto:finaclesupport@edgeverve.com)