APAR: IV07674
Symptom: Expose recertification policy APIs.
Note: Provides the ability to search, add, modify, execute and delete recertification policies in IBM Tivoli Identity Manager from a remote application.
The following classes/interfaces are exposed to provide recertification policy management capabilities through APIs.

1. Core classes:
   - com.ibm.itim.apps.policy.RecertificationPolicyManager
   - com.ibm.itim.apps.policy.RecertificationPolicyMO
   - com.ibm.itim.dataservices.model.policy.recert.RecertificationPolicy
2. Dependent classes:
   - com.ibm.itim.dataservices.model.policy.recert.RecertificationParticipant
   - com.ibm.itim.dataservices.model.policy.RoleTarget
   - com.ibm.itim.dataservices.model.policy.GroupTarget
   - com.ibm.itim.dataservices.model.policy.ServiceTarget
   - com.ibm.itim.scheduling.RecurringTimeSchedule
3. Abstract classes extended by RecertificationPolicy indirectly/directly:
   - com.ibm.itim.dataservices.model.policy.DirectoryPolicy
   - com.ibm.itim.dataservices.model.policy.ScopedPolicy
   - com.ibm.itim.dataservices.model.policy.ServicePolicy
4. Interface implemented by RecertificationPolicy or dependent classes indirectly/directly:
   - com.ibm.itim.dataservices.model.policy.Policy
   - com.ibm.itim.dataservices.model.policy.IPolicyTarget
   - com.ibm.itim.scheduling.Schedulable

The javadocs for the above mentioned classes will be available in $ITIM_HOME/extensions/5.1/api/index.html file.

Please refer the $ITIM_HOME/extensions/5.1/doc/APAR_Doc/IV07674.zip file for the examples on the usage of recertification policy APIs.

Limitations of the current implementation of recertification policy API:

1. Recertification policy with advance workflow definition capabilities cannot be created using the API. The advance workflow of the recertification policies which are already defined from the ITIM console UI cannot be modified using the API. The API cannot be used to convert a simple recertification policy to an advanced one.

2. Default Notification templates are set to the created recertification policy based on the recertification policy type and reject action set in the recertification policy object. The notification templates set to the existing recertification policy cannot be modified using API.

3. Recertification policies cannot be searched using the SearchMO API.

4. The recertification policy defined with advance workflow can be searched using the exposed APIs but invoking the following 'get' methods on the RecertificationPolicy object (to get the value of attributes specific to the recertification policy workflow) may return 'null' or may not return correct value.

a. getRecertificationParticipant()
b. getRejectAction()
c. getRejectionNotificationParticipant()
d. getTimeoutAction()
e. getTimeoutPeriod()

The following set methods cannot be used to update the existing recertification policies that are defined with advance workflow.

a. setRecertificationParticipant(RecertificationParticipant recertifier)
b. setRejectAction(java.lang.String rejectAction)
c. setRejectionNotificationParticipant(RecertificationParticipant participant)
d. setTimeoutAction(java.lang.String timeoutAction)
e. setTimeoutPeriod(int period)

5. The following methods are inherited methods but not supported for the recertification policy.

a. setAccessRights(java.util.Collection rights)
b. setAuthorizationOwners(java.util.Collection owners)