Change Tracking API

This chapter contains the following sections:

Change Tracking API

You can use the Change Tracking API to track changes that are made through their module and to record the changes in the database.

The constructor is ChangeTrackingAPI.

```
package com.cloupia.feature.foo.scheduledTasks;

import org.apache.log4j.Logger;

import com.cloupia.feature.foo.FooModule;
import com.cloupia.model.cIM.ChangeRecord;
import com.cloupia.service.cIM.inframgr.AbstractScheduleTask;
import com.cloupia.service.cIM.inframgr.FeatureContainer;
import com.cloupia.service.cIM.inframgr.cmdb.ChangeTrackingAPI;

/**
 * This is a simple example demonstrating how to implement a scheduled task.  This task is
 executed
 * every 5 mins and simply makes a logging statement and increments the number of times
it's been
 * executed.  It removes itself from the system once it has been executed twice.  It also
 * demonstrates how you can use the change tracking APIs to track changes made to the system.

 *
 */
public class DummyScheduleTask extends AbstractScheduleTask {

 private static Logger logger = Logger.getLogger(DummyScheduleTask.class);

 private int numTimesExecuted = 0;

 private static final long TWO_MINS = 60*1000*2;
 private static final int MAX_TIMES_EXECUTED = 2;

 public DummyScheduleTask() {
  super("foo");
 }

 @Override
 public void execute(long lastExecution) throws Exception {
  logger.info("vxvxvxvxvx - dummyTask has been executed " + numTimesExecuted + " times.");
  numTimesExecuted++;

  if (numTimesExecuted == MAX_TIMES_EXECUTED) {
   logger.info("vxvxvxvxvx - removing dummyTask");
   FooModule module = (FooModule) FeatureContainer.getInstance().getModuleById("foo");
   //NOTE: Use getTaskName() and NOT getScheduleTaskName(), it's really important
   //We distinguish the two: getTaskName is used internally by the system, where we do
   //some extra stuff to ensure uniqueness of the task name (prepend moduleID), so we need
 to
   //make sure to use this when removing tasks!
   module.removeScheduleTask(this.getTaskName());
   //use the static ChangeTrackingAPI to create an instance of ChangeRecord, these are just
 values you'd like have
   //tracked and store in the changes DB
   ChangeRecord rec = ChangeTrackingAPI.create("openAutoDeveloper",
ChangeRecord.CHANGE_TYPE_DELETE, "Dummy Task removed from System",
                 "foo dummy task");
   //insert the record like so
   ChangeTrackingAPI.insertRecord(rec);
  }
 }

 @Override
 public long getFrequency() {
  return TWO_MINS;
 }

 @Override
 protected String getScheduleTaskName() {
  //usually good idea to name your task something descriptive
  return "dummyTask";
 }

}
```

Tip To view the change tracking records (CMDB) from the Cisco UCS Director GUI, choose **Administration > Integration > Change Records**.