**com.sterlite.cal.brm.core.repository.
IBRMRepository**

+ Object getResultByQuery
  (Map< String, Object >
  parameterMap, QueryConfig
  queryConfig)
+ Map< String, Object
  > queryForMapForSingleRecord
  (Map< String, Object > parameterMap,
  QueryConfig queryConfig)
+ List< Map< String,
  Object > > queryForListMultiple
  Records(Map< String, Object
  > paramMap, QueryConfig queryConfig)
+ List< Map< String,
  Object > > queryForListMultiple
  Records(Map< String, Object
  > paramMap, String queryToBeExecuted,
  String dataSource)
+ void updateByQuery
  (Map< String, Object
  > parameterMap, QueryConfig
  queryConfig)
+ public< T > T getObjectBy
  Query(Map< String, Object
  > parameterMap, QueryConfig
  queryConfig, RowMapper< T >
  rowMapper)
+ public< T > List< T
  > getListByQuery(Map
  < String, Object > parameter
  Map, QueryConfig queryConfig,
  RowMapper< T > rowMapper)

**com.sterlite.cal.core.base.restclient.
BaseRestClient**

~ public< T > T getForObject
  (String uri, Map< String,
  String[]> parameterMap,
  Class< T > responseType,
  Map< String, String > requestHeader
  Parameters)
~ public< T > T deleteForObject
  (String uri, Map< String,
  String[]> parameterMap, Class
  < T > responseType, Map< String,
  String > requestHeaderParameters)
final< T > T postForObject
  (String uri, Object request,
  Class< T > responseType, Map
  < String, String > requestHeaderParameters)
final< T > T patchForObject
  (String uri, Object request,
  Class< T > responseType, Map
  < String, String > requestHeaderParameters)
final< T > T putForObject
  (String uri, Object request,
  Class< T > responseType,
  Map< String, String > requestHeader
  Parameters)
final< T > T deleteForObject
  (String uri, Object request,
  Class< T > responseType, Map
  < String, String > requestHeaderParameters)

**com.sterlite.cal.core.base.util.
JsonUtils**

+ String getJsonFromObject
  (Object data)
~ public< T > T getObjectFrom
  Json(String jsonData, Class
  < T > beanType)
~ public< T > T getObjectFrom
  Json(String jsonData, JavaType
  javaType)
~ public< T > T getObjectFrom
  Json(String jsonData, TypeReference
  < T > typeReference)
~ public< T > T convertValue
  (Object fromValue, TypeReference
  < T > typeReference)

Environment

~brmRepository

~jsonutil

~env

**com.sterlite.cal.core.logger.
ILogger**

+ void info(String strMethod
  Name, LogType logType,
  String strMessage)
+ void info(String strMethod
  Name, LogType logType,
  String ...strMessages)
+ void info(String strMethod
  Name, LogType logType,
  Object ...strMessage)
+ void debug(String strMethod
  Name, LogType logType, String
  strMessage)
+ void debug(String strMethod
  Name, LogType logType, String...
  strMessage)
+ void debug(String strMethod
  Name, LogType logType, Object
  ...strMessage)
+ void error(String strMethod
  Name, LogType logType, Throwable
  throwable)
+ void startLog(String
  strMethodName, LogType
  logType)
+ void endLog(String
  strMethodName, LogType
  logType)
+ void warn(String strMethod
  Name, LogType logType,
  String strMessage)
+ boolean isTraceEnabled()
+ boolean isDebugEnabled()
+ boolean isInfoEnabled()
+ boolean isWarnEnabled()
+ boolean isErrorEnabled()
+ static ILogger getInstant
  (Class<?> className)

**com.sterlite.cal.dNotification.util.
DNoticiationUtil**

+ List< Map< String,
  Object > > getBAFromCA
  (String accountNumber)
+ List< Map< String,
  Object > > getSIFromBA
  (String accountNumber)
+ List< Map< String,
  Object > > getSAFromCA
  (String accountNumber)
+ String getURL(String
  requestURI)
+ String getProperty
  (String key)
+ Map< String, String
  > prePareHeaderMap
  (String authTypeId, Map
  < String, String > header)
+ Object getObject(String
  className)
+ String getDnotificationToken()
~ public< T > T postForObject
  (String uri, Object request,
  Class< T > responseType, String
  authTypeId, Map< String, String
  > header, String operationAlias)
~ public< T > T getForObject
  (String uri, Map< String,
  String[]> parameterMap,
  Class< T > responseType,
  String authTypeId, Map< String,
  String > header, String operationAlias)

**com.sterlite.cal.dNotification.integration.
IDNotificationIntegrationService**

+ CalBaseResponse getAccount
  HistoryFromDNotification
  (String fromDate, String
  toDate, String accountNumber,
  String eventAlias, String channelAlias,
  String status)

~env

+LOGGER   ~dNotificationUtil

**com.sterlite.cal.dNotification.integration
Impl.DNotificationIntegrationImpl**

+ static final String
  MODULE

+ CalBaseResponse getAccount
  HistoryFromDNotification
  (String fromDate, String
  toDate, String accountNumber,
  String eventAlias, String channelAlias,
  String status)