Covidien Gateway Remote Service

Controller Interface Specifications

**Background**

The **Controller** (aka PC Agent Interface) is limited by its relationship to the **Remote Service Agent** (aka Laptop Agent). In particular, the communications pattern is asynchronous, as seen in the images below. Consequently the Controller Interface Specifications are written to adapt to the situation. There are two viable styles of how the **UI** ( User Interface) might interface with the Controller.

In the first style, the UI passes a **delegate** (aka function pointer) callback to be invoked when Controller completes its work. In this case, the Controller creates a transaction id (X-ID), passes the request to the RSA, and waits for a response which includes the X-ID of interest. Upon receipt, the Controller invokes the delegate to convey the information to the UI.

Delegate()  
**Upcall**

REQ  
(delegate)

REQ (X-ID)

RESP  
(X-ID)

CTRL

RSA

UI

In the second style, the UI makes two (or more) calls to the Controller. The first call establishes the request and returns a unique transaction id (X-ID). The second (and subsequent) calls are a form of polling, where the UI makes the equivalent request of the Controller, passing the X-ID as a parameter, hoping for a non-null response.

CTRL

RSA

UI

REQ (X-ID)

RESP  
(X-ID)

REQ, X-ID  
RESP

REQ  
X-ID

Both styles are supported by these specifications. The result is a common pattern of API specifications.

1. Original call with appropriate request parameters, which returns a response of a unique transaction id.
2. Polling call with the unique transaction id as a parameter, which returns NULL in the case of a response not yet being available from the server OR a response already reaped and discarded. An additional output parameter is provided so that the reason for the NULL response may be recognized. The reasons are enumerated as “No Response From Server”, “Server Response” and “Response Previously Reaped”.
3. Original call with appropriate request parameters plus a callback function (e.g., delegate). Upon response from the server, the callback function (if not NULL) will be invoked. TBD - Potential design evolution may allow an additional parameter to be passed for callback data, but anonymous function design may obviate this need.

**Interface Methods / Functions**

|  |  |  |
| --- | --- | --- |
| Functionality | Parameters | Results |
| Login | UserId, Password | Boolean – true means success |
| Logout | none | none |
|  |  |  |
| GetDevices | [ Filters ] | TransactionId |
| GetDevices | TransactionId | IEnumerable < Device Guids >  Enum ResponseStatus |
| GetDevices | [ Filters ], GetDevicesDelegate | Delegate( IEnumerable < Device Guids > ) |
|  |  |  |
| GetDeviceStatus | Device Guid | TransactionId |
| GetDeviceStatus | TransactionId | IEnumerable < KeyValuePair< key, value > >  Enum ResponseStatus |
| GetDeviceStatus | Device Guid, GetDeviceStatusDelegate | Delegate( IEnumerable < KeyValuePair< key, value > > ) |
|  |  |  |
| GetHeaders | [ Filters ] | TransactionId |
| GetHeaders | TransactionId | IEnumerable < Header >  Enum ResponseStatus |
| GetHeaders | [ Filters ], GetHeadersDelegate | Delegate( IEnumerable < Header > ) |
|  |  |  |
| GetNotification | Notification Guid | TransactionId |
| GetNotification | TransactionId | Notification ?  Enum ResponseStatus |
| GetNotification | Notification Guid, GetNotificationDelegate | Delegate( Notification ) |
|  |  |  |
| PostNotification | Notification Guid | TransactionId |
| PostNotification | TransactionId | Boolean ? – true means success  Enum ResponseStatus |
| PostNotification | Notification, PostNotificationDelegate | Delegate( Boolean ) |
|  |  |  |
| DeleteNotification | Notification Guid | TransactionId |
| DeleteNotification | TransactionId | Boolean ? – true means success  Enum ResponseStatus |
| DeleteNotification | Notification, DeleteNotificationDelegate | Delegate( Boolean ) |
|  |  |  |
| UndeleteNotification | Notification Guid | TransactionId |
| UndeleteNotification | TransactionId | Boolean ? – true means success  Enum ResponseStatus |
| UndeleteNotification | Notification, UndeleteNotificationDelegate | Delegate( Boolean ) |
|  |  |  |
| UpdateNotification | Notification Guid | TransactionId |
| UpdateNotification | TransactionId | Boolean ? – true means success  Enum ResponseStatus |
| UpdateNotification | Notification, UpdateNotificationDelegate | Delegate( Boolean ) |
|  |  |  |
| ExpungeNotification | Notification Guid | TransactionId |
| ExpungeNotification | TransactionId | Boolean ? – true means success  Enum ResponseStatus |
| ExpungeNotification | Notification, ExpungeNotificationDelegate | Delegate( Boolean ) |