Subscription

Modules can subscribe to events from other modules.

A client subscribes to an event from a host. When the host event occurs, it will check for client subscriptions and will send data to each of the subscribers.

To subscribe for an event, the client will send a subscription request to the host. This request will provide:

* The Q of the subscriber
* A prototype of the Q element to be used for sending the data.
  + This has the Cmd, Sub, and Data2 filled with the necessary info.
  + Data1 is reserved for the data of the subscribed event.

The

When a host experiences an event that is subscribe, it will check id there are subscriptions and if there are, it will send the data to each subscriber. The subscription prototype queue element will be sued and Data1 will be populated with data. The subscriber must be prepared to receive the data type that is sent by the subscriber in Data1.

The KATE library has a Subscription class. The following VIs are used to process subscription activities:

* Subscr.Subscribe to Event
  + Used by a client.
  + Sends a request to the Host to subscribe for data.
  + The request will contain the Q of the client and a prototype queue element for sending the event data.
  + This request also has a Disable input. This is false by default which results in the subscription being added. If it is set to true, the any existing subscription will eb canceled.
* Subscr.Process Subscription Request.vi
  + Used by a host.
  + Receives a request subscribe data.
  + If the Disable is true, any existing subscription will eb canceled. If it is false, the subscription will eb added.
* Subscr.Service Subscriptions.vi
  + Used by a host.
  + When an event occurs that is available for subscriptions, it will check if a client has subscribed to that event. If so, then it will send data to the client. If not, then it will do nothing.

Clients that support Subscription must have a ‘Subscribe’ case to receive subscriptions.