

How to use Spectra Application Listener callbacks

- [Using callbacks during the recovery.](#)
- [How MOEX Spectra Market Data Adapter reacts on Sequence Reset message](#)

Using callbacks during the recovery.

The recovery process is started when there is inconsistent Market Data for a particular Instrument. At the start of the recovery process, the following callback is fired:

B2BITS_SpectraApplicationListeners.h

```
/// Fired when recovery is started
/// @param subsItem Description of the instrument
virtual void InstrumentListener::onRecoveryStarted( SpectraSubscriptionItem const& subsItem );
```

During the recovery onSnapshot and onIncrement callbacks are fired to provide a user with the most recent Market Data that the Adapter has and the data can be applied:

B2BITS_SpectraApplicationListeners.h

```
/// Fired when update for instrument is available.
/// @param subsItem Description of the instrument
/// @param msgs Increment container with group entry inside Market Data - Incremental Refresh message assigned
to instrument.
/// @param incrementsCount Increments count into Msgs
/// @return false if book need start recovery, otherwise true
/// @note it is possible that last message in msgs contains LastFragment==0
/// (in case when FIX message with LastFragment==1 doesn't contain updates for this subsItem)
virtual bool InstrumentListener::onIncrement( SpectraSubscriptionItem const& subsItem, Engine::FIXMessage
const* const* msgs, size_t incrementsCount );

/// Fired when user should reset book with the bnew values.
/// @param subsItem Description of the instrument
/// @param msgs Snapshots container
/// @param snapshotsCount Snapshots count into Msgs
virtual void InstrumentListener::onSnapshot( SpectraSubscriptionItem const& subsItem, Engine::FIXMessage const*
const* msgs, size_t snapshotsCount);
```

But the Market Data can't be treated as actual on the Exchange.

The Market Data for an instrument can be considered as actual only after onRecoveryStopped callback:

B2BITS_SpectraApplicationListeners.h

```
/// Fired when recovery is ended
/// @param subsItem Description of the instrument
/// @param reason Reason of recovery
virtual void InstrumentListener::onRecoveryStopped( SpectraSubscriptionItem const& subsItem, RecoveryReason
reason );
```

It means that the recovery process has been done and Market Data is in a consistent state.

How MOEX Spectra Market Data Adapter reacts on Sequence Reset message

The new type of messages (a message 'Sequence Reset') was introduced in Spectra FAST protocol specification v 1.4.2.

MOEX Spectra Market Data Adapter handles Sequence Reset message in the following way:

1. A Sequence Reset message received in Incremental feed initiates recovery mechanism for all subscribed instruments.
In this case, callbacks `Spectra::InstrumentListener::onRecoveryStarted()` and `Spectra::InstrumentListener::onBookReset()` are fired.
2. Sequence filtering mechanism isn't applied to Sequence Reset messages.
Therefore, `OnRecoveryStarted` and `OnBookReset` callbacks would be fired for each sequence reset message (i.e. twice due to A-B arbitrage).