SuperNetwork

Generated by Doxygen 1.8.16

Chapter 1

Mantis Vision: SuperNetwork

A plugin providing services/filters for transmitting and receiving MVX data over network.

Table of Contents

Details

Compiled with MVZMQNetwork version 5.0.0.

Chapter 2

Release Notes

1.0.0

Initial version.

Plugin

- **1.0.0_P1** | renamed TargetSuperNetworkTransmitter's **"Dropped frames count"** and **"Reset dropped frames counter"** parameters to **"Dropped atoms count"** and **"Reset dropped atoms counter"** respectively

Documentation

- **1.0.0_D1** | added 'release notes' section
- **1.0.0_D2** | added a section for each plugin's filter

Build support

- **1.0.0_BS1** | introduced config files specifying collection of all files (libraries, assets, ...) associated with the plugin - one MVX2File.cfg file per each combination of OS, architecture and build configuration, located next to the plugin file (dll/so/...)

2.0.0

Plugin

- **2.0.0_P1** | updated for `Mvx2` framework version 3.0.0
- **2.0.0_P2** | updated `MVZMQNetwork` dependency to version 2.0.0
- **2.0.0_P3** | **"Send buffer capacity"** parameter of TargetSuperNetworkTransmitter is interpreted as buffer capacity 'per peer and per stream' instead of 'per peer' (i.e. number of streams in frames affects buffer capacity as well)
- **2.0.0_P4** | **"Receive buffer capacity"** parameter of SourceSuperNetworkReceiver is interpreted as buffer capacity 'per stream' instead of 'per connection' (i.e. number of streams in frames from transmitter affects buffer capacity)
- **2.0.0_P5** | **"Receive buffer capacity"** parameter of MutateSuperNetworkReceiver is interpreted as buffer capacity 'per stream' instead of 'per connection' (i.e. number of streams in frames from transmitter affects buffer capacity)
- **2.0.0_P6** | **"Receive buffer capacity"** parameter of SourceSuperNetworkMultiReceiver is interpreted as buffer capacity 'per transmitter and per stream' instead of 'per transmitter' (i.e. number of streams in frames from transmitter affects buffer capacity)

Build support

- **2.0.0_BS1** | size of Android and LuminOS libraries reduced by ~90%
- **2.0.0_BS2** | android API level raised from 19 to 21
- **2.0.0_BS3** | Linux and MacOS binaries do not consist of a versioned plugin file and a version-neutral symlink file anymore - the plugin file itself has version-neutral name

3.0.0

Plugin

- **3.0.0_P1** | updated for `Mvx2` framework version 4.0.0
- **3.0.0_P2** | updated `MVZMQNetwork` dependency to version 3.0.0

4.0.0

Plugin

- **4.0.0_P1** | updated `MVZMQNetwork` dependency to version 3.1.0 (experimental IPv6 support)
- **4.0.0_P2** | format of **"Address List"** parameter in SourceSuperNetworkMultiReceiver is stricter: all 4 elements (protocol, address, commands port and data port) are mandatory
- **4.0.0_P3** | introduced **"Enable IPv6"** parameter to TargetSuperNetworkTransmitter
- **4.0.0_P4** | introduced **"Enable IPv6"** parameter to SourceSuperNetworkReceiver
- **4.0.0_P5** | introduced **"Enable IPv6"** parameter to MutateSuperNetworkReceiver
- **4.0.0_P6** | introduced **"Enable IPv6"** parameter to SourceSuperNetworkMultiReceiver

4.1.0

Plugin

- **4.1.0_P1** | updated `MVCommon` 3rdparty dependency to version 3.0.0
- **4.1.0_P2** | updated for `Mvx2` framework version 5.0.0
- **4.1.0_P3** | updated `MVZMQNetwork` 3rdparty dependency to version 4.0.0
- **4.1.0_P4** | logs generated by the plugin use custom `SuperNetwork` tag instead of the generic (`MVX2`) one

4.2.0

Plugin

- **4.2.0_P1** | updated `MVCommon` 3rdparty dependency to version 4.0.0
- **4.2.0_P2** | updated for `Mvx2` framework version 6.0.0
- **4.2.0_P3** | updated `MVZMQNetwork` 3rdparty dependency to version 5.0.0

Documentation

- **4.2.0_D1** | introduced PDF documentation as an alternative to the HTML one:
  - `doc/SuperNetwork.pdf`

Chapter 3

Source filters

3.1 SourceSuperNetworkReceiver

A source filter for reception of processed frames via network.

The reception is based on MVZMQNetwork protocol (which is based on ZeroMQ networking library).

Type GUID

631F7270-CC7F-4BA0-83F5-02AD18E47D5B

Parameters

- **"Enable IPv6"**
  Enables IPv6 support. By default the IPv6 support is disabled and any attempt to use an IPv6 address fails.
- **"Source commands address"**
  An address in zmq-accepted format of commands socket of a transmitter to connect to (e.g. 'tcp://192.168.↩0.11:5555', 'tcp://localhost:5555', 'tcp://fe80::cdf9:339a:8ff1:5f17:5555')
- **"Source data address"**
  An address in zmq-accepted format of data socket of a transmitter to connect to (e.g. 'tcp://192.168.0.11↩:5556', 'tcp://localhost:5556', 'tcp://fe80::cdf9:339a:8ff1:5f17:5556').
- **"Receive buffer capacity"**
  A capacity of receive data buffer (in messages) per stream (check ZMQ_RCVHWM option of zmq_↩setsockopt()).
- **"Response receive timeout"**
  An interval available for establishing connection with a transmitter (in ms) before the operation is considered unsuccessful. In such case the filter remains uninitialized.
- **"Unsupported transmitter protocol versions"**
  A comma-separated list of transmitter versions the receiver shall not communicate with.

3.2 MutateSuperNetworkReceiver

A mutate filter for reception of processed frames via network. Since the filter is not a source filter, it allows to receive data further in a processing pipeline, which has its own actual source. The filter thus operates with two different sources of data: a pipeline and a network. The principle of merging the two sources of data is:

1. atom received from a pipeline is cloned into a resulting atom,
2. new data layers received from a network are added to the resulting atom,
3. data layers received from a network, which are present also in the atom received from a pipeline, override them. Presence/equivalence of data layers is checked using their data profiles.

The reception is based on MVZMQNetwork protocol (which is based on ZeroMQ networking library).

Type GUID

1831CD42-758E-44C4-AF5F-C56A25E01EAC

Parameters

- **"Enable IPv6"**
  Enables IPv6 support. By default the IPv6 support is disabled and any attempt to use an IPv6 address fails.
- **"Source commands address"**
  An address in zmq-accepted format of commands socket of a transmitter to connect to (e.g. 'tcp://192.168.↩0.11:5555', 'tcp://localhost:5555', 'tcp://fe80::cdf9:339a:8ff1:5f17:5555')
- **"Source data address"**
  An address in zmq-accepted format of data socket of a transmitter to connect to (e.g. 'tcp://192.168.0.11↩:5556', 'tcp://localhost:5556', 'tcp://fe80::cdf9:339a:8ff1:5f17:5556').
- **"Receive buffer capacity"**
  A capacity of receive data buffer (in messages) per stream (check ZMQ_RCVHWM option of zmq_↩setsockopt()).
- **"Response receive timeout"**
  An interval available for establishing connection with a transmitter (in ms) before the operation is considered unsuccessful. In such case the filter remains uninitialized.
- **"Unsupported transmitter protocol versions"**
  A comma-separated list of transmitter versions the receiver shall not communicate with.

3.3 SourceSuperNetworkMultiReceiver

A source filter for reception of processed frames via network from multiple independent transmitters.

The reception is based on MVZMQNetwork protocol (which is based on ZeroMQ networking library).

Inability to connect to any of specified transmitters does not prevent the filter from communicating with other transmitters.

There are multiple rules that each transmitter must follow (the rules are checked by the receiver) in order to not be excluded from the communication:

1. all transmitters must use exactly the same data profile (the first successfully connected transmitter's profile is used as a reference),
2. all transmitters must provide unique stream IDs, i.e. no two transmitters provide the same stream ID data,
3. transmitters are allowed to provide multiple stream IDs, but all must provide the same number of stream IDs,
4. the receiver declares the same number of frames as the one transmitter, which declares the highest number of frames.

Type GUID

1C36A76F-516A-490A-8287-BB7E11610FF0

Parameters

- **"Receive buffer capacity"**
  A capacity of receive data buffer (in messages) per transmitter and per stream (check ZMQ_RCVHWM option of zmq_setsockopt()).
- **"Response receive timeout"**
  An interval available for establishing connection with a transmitter (in ms) before the operation is considered unsuccessful. In such case the transmitter is excluded from the communication.
- **"Enable IPv6"**
  Enables IPv6 support. By default the IPv6 support is disabled and any attempt to use an IPv6 address fails.
- **"Address List"**
  A whitespace-separated list of commands and data sockets of all transmitters to connect to. Expected format for addresses is:
  - `PROTOCOL://ADDRESS:CMD_PORT:DATA_PORT`

  where
  - PROTOCOL is one of zmq-supported transport protocols (tcp, udp, ...),
  - ADDRESS is a network address of a transmitter to connect to (IPv4, IPv6 or host name),
  - CMD_PORT and DATA_PORT are port numbers for command and data sockets respectively.

  Example of valid values:
  - `tcp://127.0.0.1:1234:4321 tcp://remotehost:5500:5501 tcp://fe80↩ ::cdf9:339a:8ff1:5f17%10:5854:5599`
- **"Unsupported transmitter protocol versions"**
  A comma-separated list of transmitter versions the receiver shall not communicate with.

Chapter 4

Target filters

TargetSuperNetworkTransmitter

4.1 TargetSuperNetworkTransmitter

A target filter for transmission of processed frames via network.

The transmission is based on MVZMQNetwork protocol (which is based on ZeroMQ networking library).

Type GUID

C53EDD17-1417-4D17-A69A-C0FB06C2A176

Parameters

- **"Enable IPv6"**
  Enables IPv6 support. By default the IPv6 support is disabled and any attempt to use an IPv6 address fails.
- **"Transmitter commands address"**
  An address in zmq-accepted format to bind the transmitter's commands socket with (e.g. 'tcp://∗:5555').
- **"Transmitter data address"**
  An address in zmq-accepted format to bind the transmitter's data socket with (e.g. 'tcp://∗:5556').
- **"Send buffer capacity"**
  A capacity of data buffer (in messages) per peer and per stream (check ZMQ_SNDHWM option of zmq_↩ setsockopt()).
- **"Unsupported receiver protocol versions"**
  A comma-separated list of receiver versions the transmitter shall not respond to.
- **"Connection is critical"**
  Determines whether the filter should report that there is no problem even when it actually fails to bind to commands and/or data socket. The filter in such case is unable to transmit data, but it does not prevent the rest of the pipeline from functioning.
- **"Streaming Enabled"**
  Enables/disables data transmission. Allows to manually pause transmission of data to receivers.

- **"Transmitting"**
  An indication of current data transmission state. The value is updated after each attempt to transmit an atom.
- **"Dropped atoms count"**
  A count of unsuccessfully transmitted atoms. Provides a way for monitoring the transmission reliability.
- **"Reset dropped atoms counter"**
  Resets the dropped atoms counter to 0.