

Virtual Human Messaging

Communication between most modules happens by message passing, which is implemented in [ActiveMQ](#). Messages are broadcasted through the system and a component can subscribe to certain messages. Developers can use the [VH Messaging library](#) (VHMsg) to quickly implement receiving and sending ActiveMQ messages in a variety of languages, including C++, C# and Java.

Every module has its own messaging API. In addition, each module should at least implement the basic messaging protocol that allows modules to request and share whether they are online:

- [vrAllCall](#) [reason], pings all modules.
- [vrComponent](#) *component-idsub*, response by module to 'vrAllCall'; should also be sent on start-up by module.
- [vrKillComponent](#) {*component-id/all*}, requests a single module or all modules to shut down.
- [vrProcEnd](#) *component-idsub*, sent by module on exit.

Individual messages are listed under the [Messages](#) page. Conceptual areas with groups of messages are described under the [Sub Areas](#) page.