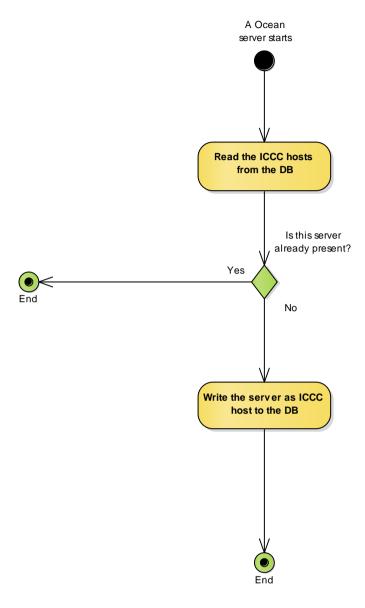
Ocean's ICCC Terminologies

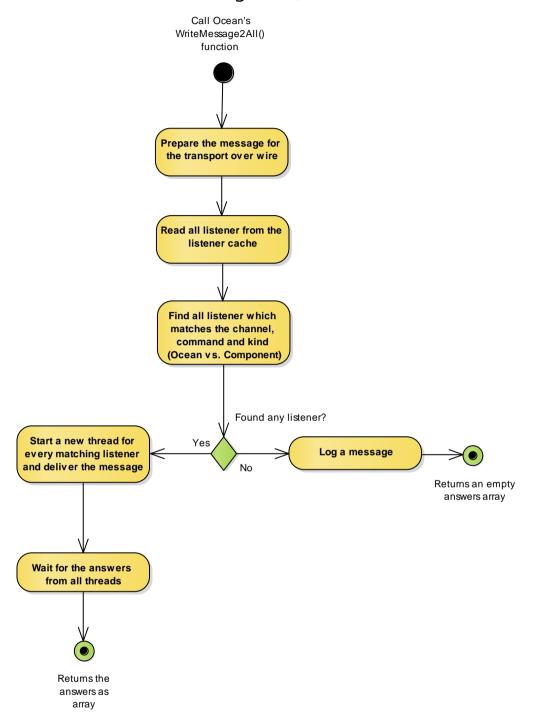
Listener := An ICCC command at a given channel which is located on any or all of the servers Host := An ICCC host is a server which provides at least one ICCC command. Thus, any host provides at least one listener.

Ocean's ICCC start-up

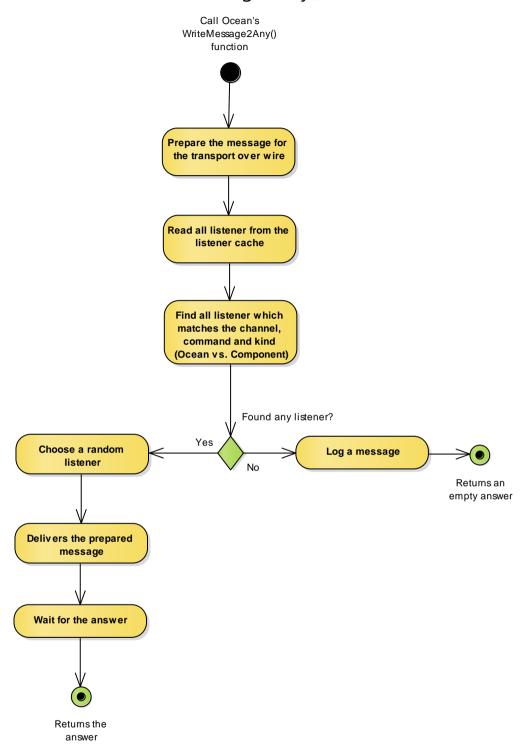


This diagram shows only the ICCC parts of Ocean's start-up process! Further, this diagram is only for Ocean true -- other external ICCC components e.g. with C#/.NET, Python, etc. are setting up differently.

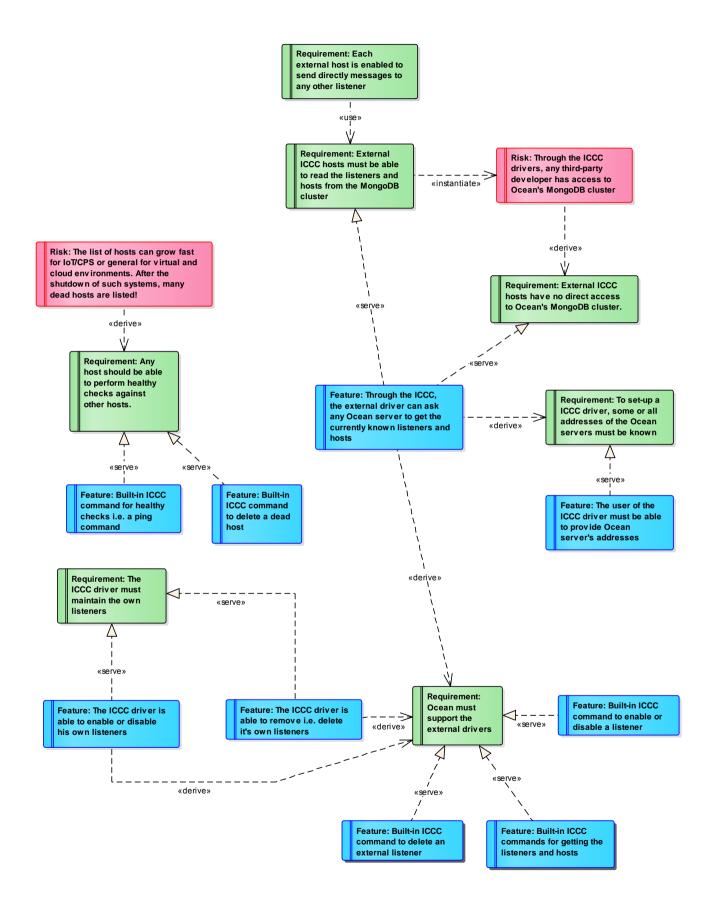
Ocean's WriteMessage2All() function



Ocean's WriteMessage2Any() function



Requirements for ICCC drivers



Class diagram for an Ocean ICCC driver

ICCC + HandlerIncomings + ICCCConnection + ICCCHost + ICCCListener + ICCCProcessor + Kind + Registrar + SystemChannels + IICCCMessage + SystemMessages



(from ICCC)



- ALL: byte = 255 {readOnly}
- Component: byte = 1 {readOnly}
 Ocean: byte = 0 {readOnly}

ICCCHost and ICCCListener are classes to maintain lists with the current available entries

ICCC::ICCCHost

- hostname: string
- ipAddressPort: string
- kind: byte

«property»

- Hostname(): string
- IPAddressPort(): string
- Kind(): byte
- «constructor»
- ICCCHost() ICCCHost(string, string)

ICCC::ICCCListener

- channel: string
- command: string
- ipAddressPort: string
- isActive: bool
- kind: byte

«property»

- Channel(): string
- Command(): string
- IPAddressPort(): string IsActive(): bool
- Kind(): byte

«constructor»

- ICCCListener()
- ICCCListener(string, string, string, bool)

ICCC::HandlerIncomings

- ProcessRequest(WebRequest, WebResponse): void
- HandlerIncomings()

ICCC::SystemChannels

- ICCC: string = "System::ICCC" {readOnly}
- LOGGING: string = "System::Logging"
- NUMGEN: string = "System::NumGen" {readOnly}
- PING: string = "System::Ping" {readOnly}
- SHUTDOWN: string = "System::Shutdown" {readOnly} STARTUP: string = "System::Startup" {readOnly}
- SYSTEM: string = "System" {readOnly}

ICCC::ICCCProcessor

- INSTANCE: ICCCProcessor {readOnly}
- maintainerThread: Thread
- convertData2Message(NameValueCollection, IICCCMessage): IICCCMessage
- convertMessage2Data(IICCCMessage): NameValueCollection
- maintainer(): void
- setConfiguration(string): void
- signMessage(NameValueCollection): NameValueCollection

«constructor»

ICCCProcessor()

ICCC::Registrar

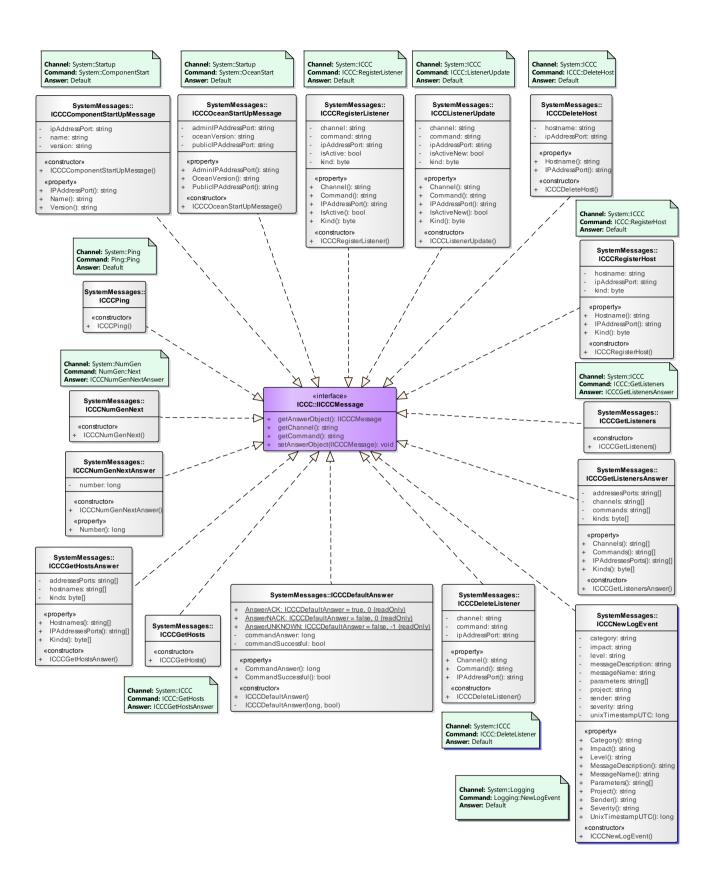
- INSTANCE: Registrar {readOnly}
- localListeners: Dictionary<string, MethodCallback
- addLocalListener(string, string, MethodCallback): void
- getLocalListeners(string, string): IEnumerable<MethodCallback

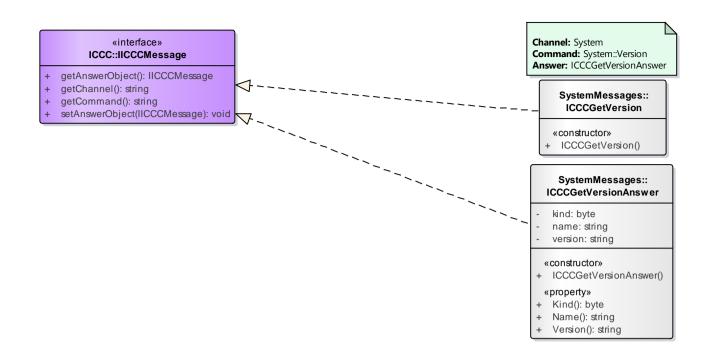
«constructor»

Registrar()

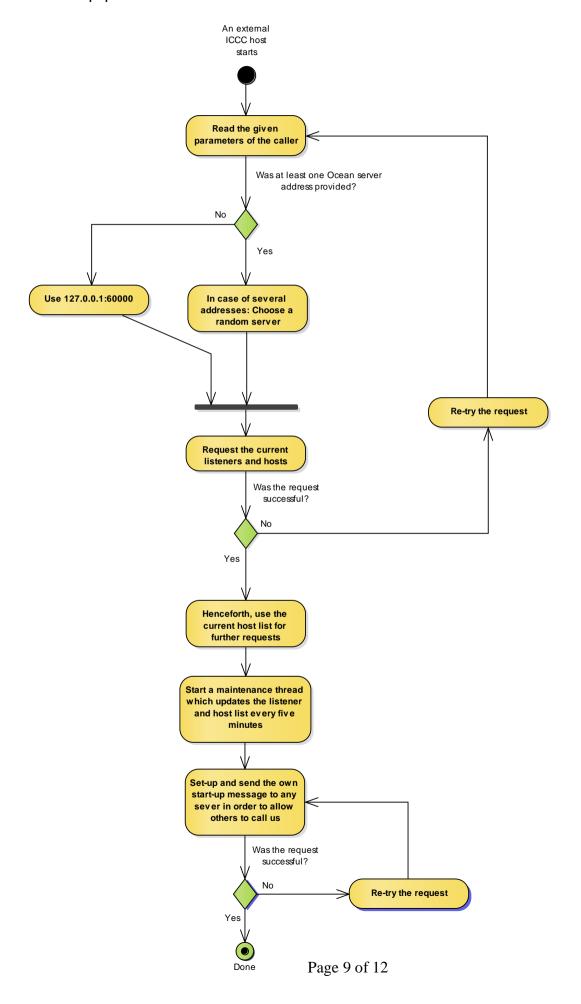
ICCC::ICCCConnection

- icccHostsCache: IEnumerable<ICCCHost>
- icccListenerCache: IEnumerable<ICCCListener>
- INSTANCE: ICCCConnection {readOnly}
- send2All(IICCCMessage, byte): IEnumerable<IICCCMessage>
- send2Any(IICCCMessage, byte): IICCCMessage
- sendMessage(NameValueCollection): void
- setConfiguration(IEnumerable<string>): void
- «constructor»

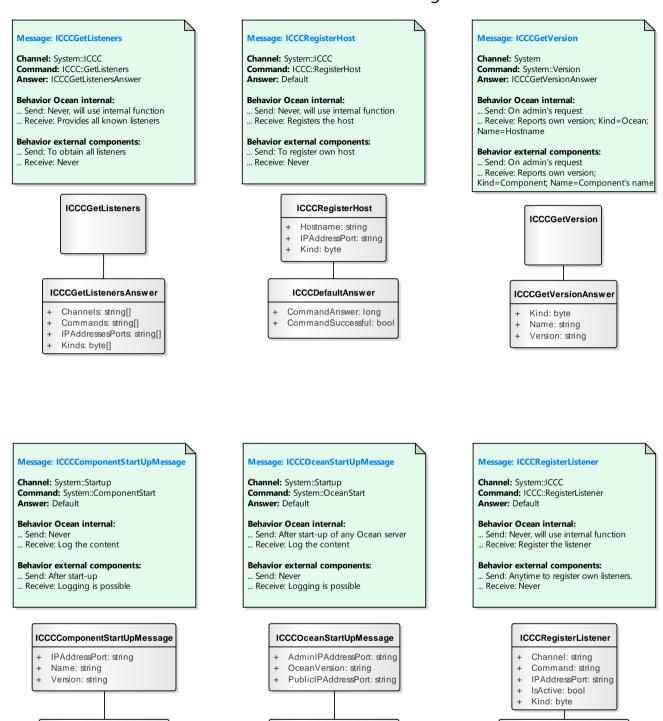




Start-up process for a external ICCC driver / host



Documentation of ICCC Messages



ICCCDefaultAnswer

CommandAnswer: long

CommandSuccessful: bool

ICCCDefaultAnswer

CommandAnswer: long

CommandSuccessful: bool

ICCCDefaultAnswer

CommandAnswer: long

CommandSuccessful: bool

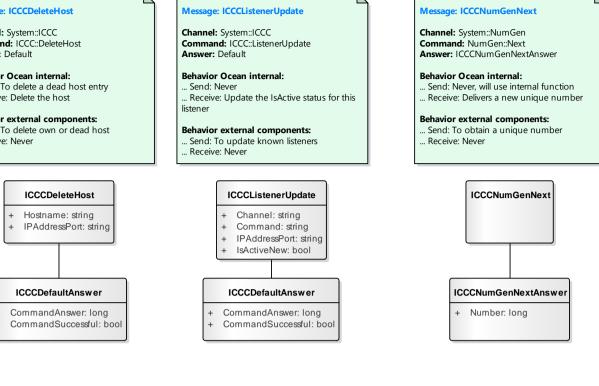
Message: ICCCDeleteHost Channel: System::ICCC Command: ICCC::DeleteHost Answer: Default Behavior Ocean internal: Send: To delete a dead host entry ... Receive: Delete the host Behavior external components: ... Send: To delete own or dead host Receive: Never **ICCCDeleteHost** Hostname: string IPAddressPort: string ICCCDefaultAnswer

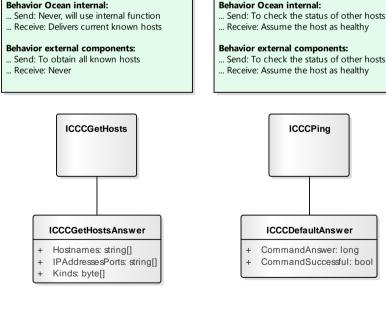
Message: ICCCGetHosts

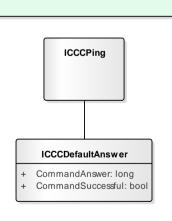
Channel: System::ICCC

Command: ICCC::GetHosts

Answer: ICCCGetHostsAnswer







. Send: To check the status of other hosts

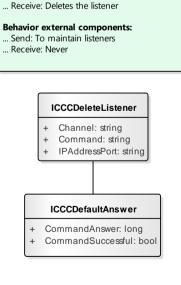
.. Receive: Assume the host as healthy

Message: ICCCPing

Channel: System::Ping

Command: Ping::Ping

Answer: Deafult



Message: ICCCDeleteListener

Command: ICCC::DeleteListener

... Send: Never, will use internal function

Channel: System::ICCC

Behavior Ocean internal:

Answer: Default

Message: ICCCNewLogEvent

Channel: System::Logging **Command:** Logging::NewLogEvent **Answer:** Default

Behavior Ocean internal:

- ... Send: Never ... Receive: Bring-in the event to the logging

Behavior external components:

- ... Send: Anytime an interesting event occurs
- ... Receive: Never

ICCCNew LogEvent

- Category: string
- + Impact: string
- Level: string
- MessageDescription: string
- MessageName: string
- Parameters: string[]
- Project: string
- Sender: string
- Severity: string
- UnixTimestampUTC: long

ICCCDefaultAnswer

- CommandAnswer: long
- CommandSuccessful: bool