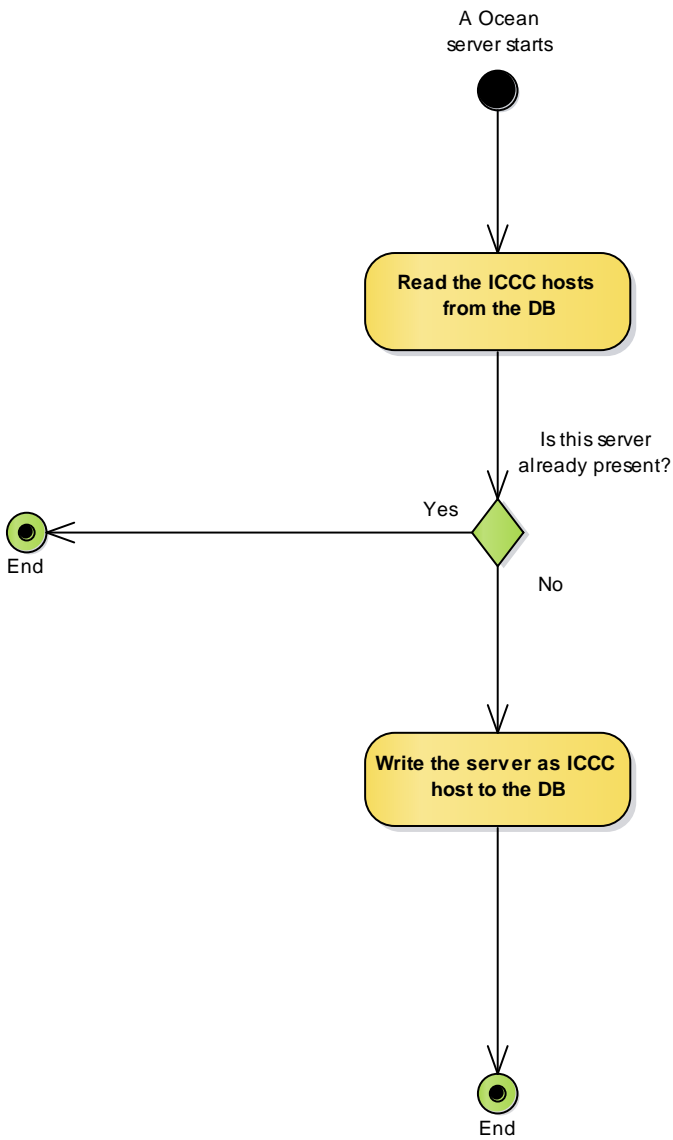


# Ocean's ICCC Terminologies

**Listener** := An ICCC command at a given channel which is located on any or all of the servers

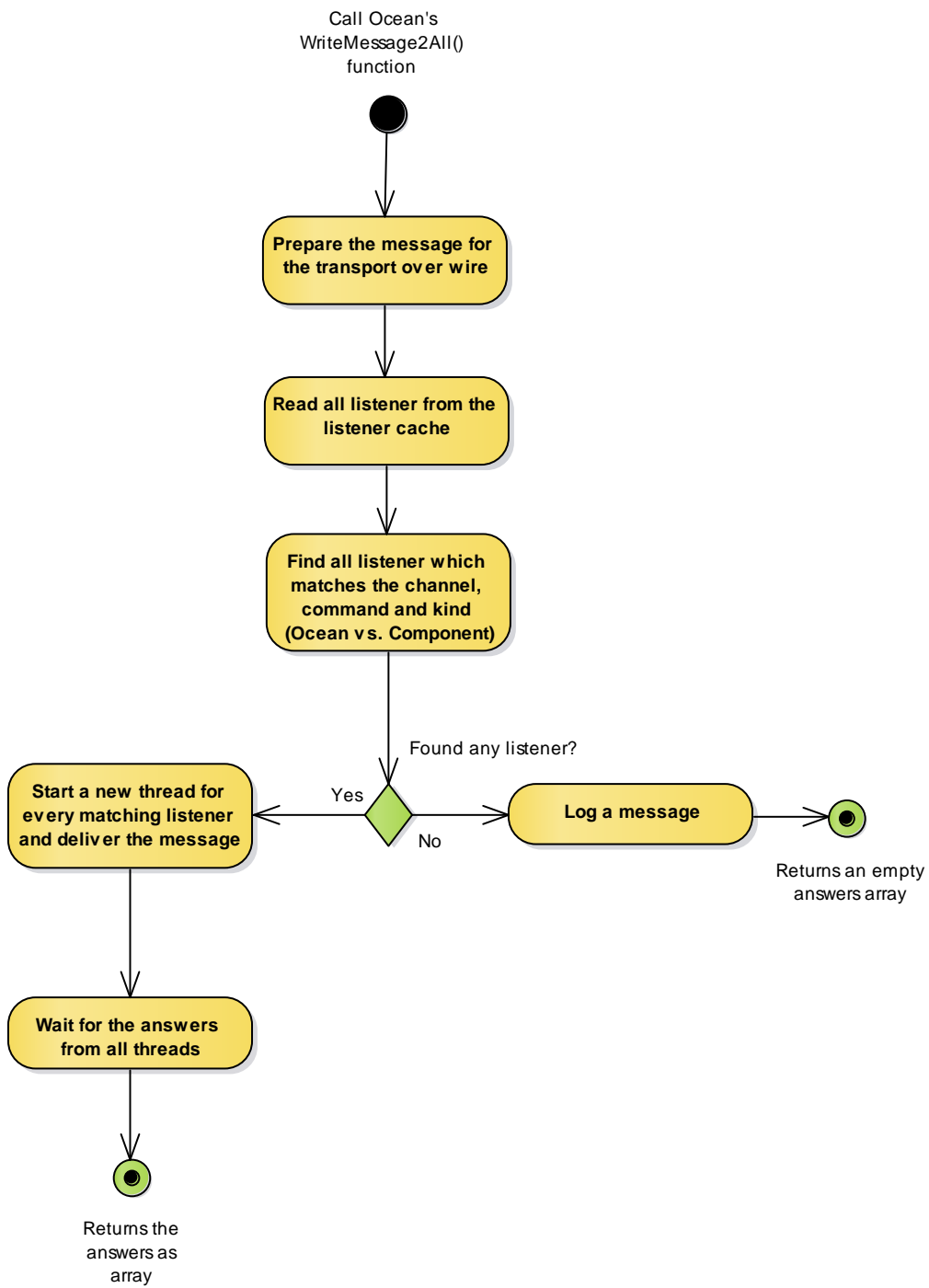
**Host** := An ICCC host is a server which provides at least one ICCC command. Thus, any host provides at least one listener.

# Ocean's ICCC start-up

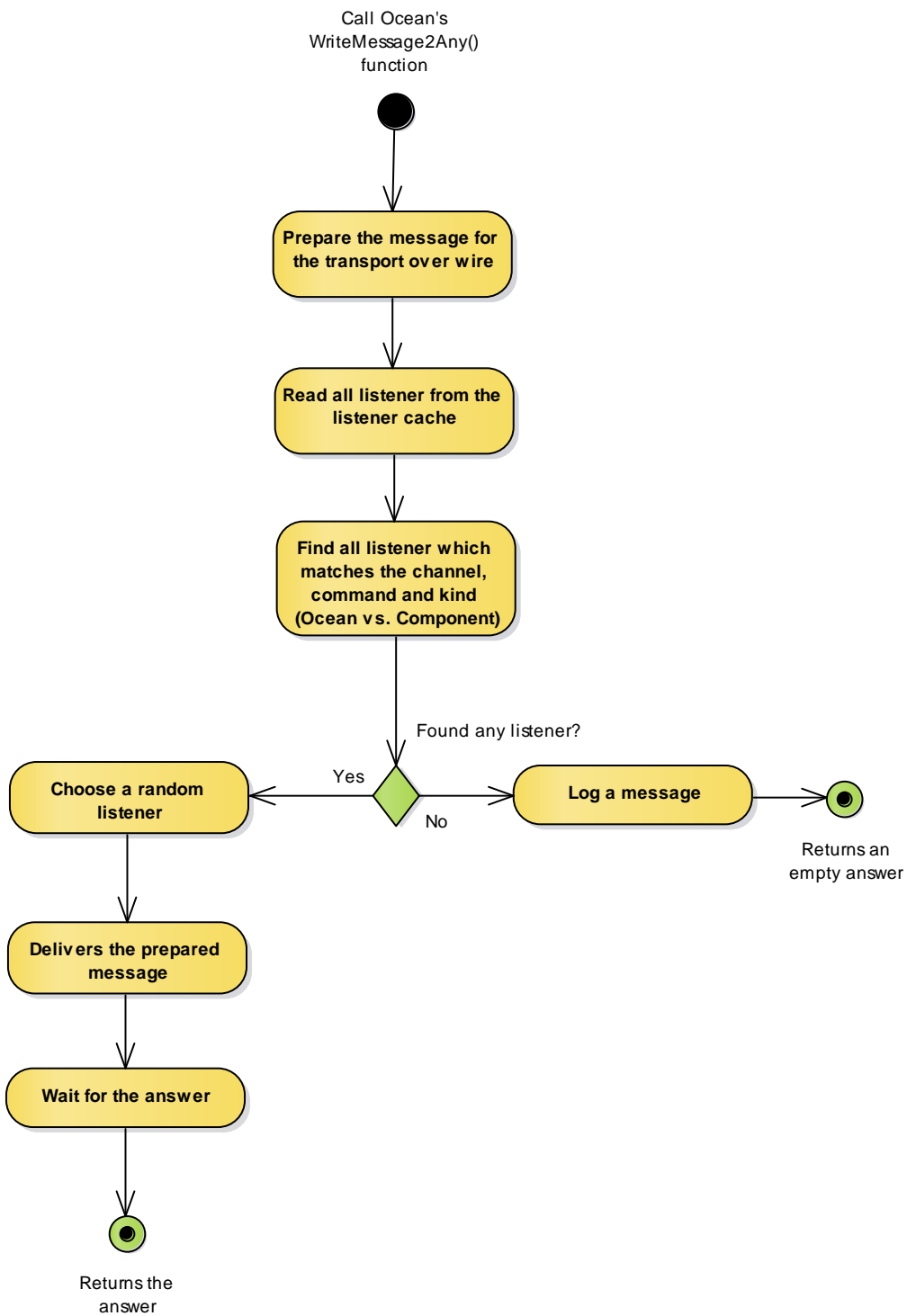


This diagram shows only the ICCC parts of Ocean's start-up process! Further, this diagram is only for Ocean true -- other external ICCC components e.g. with C#/.NET, Python, etc. are setting up differently.

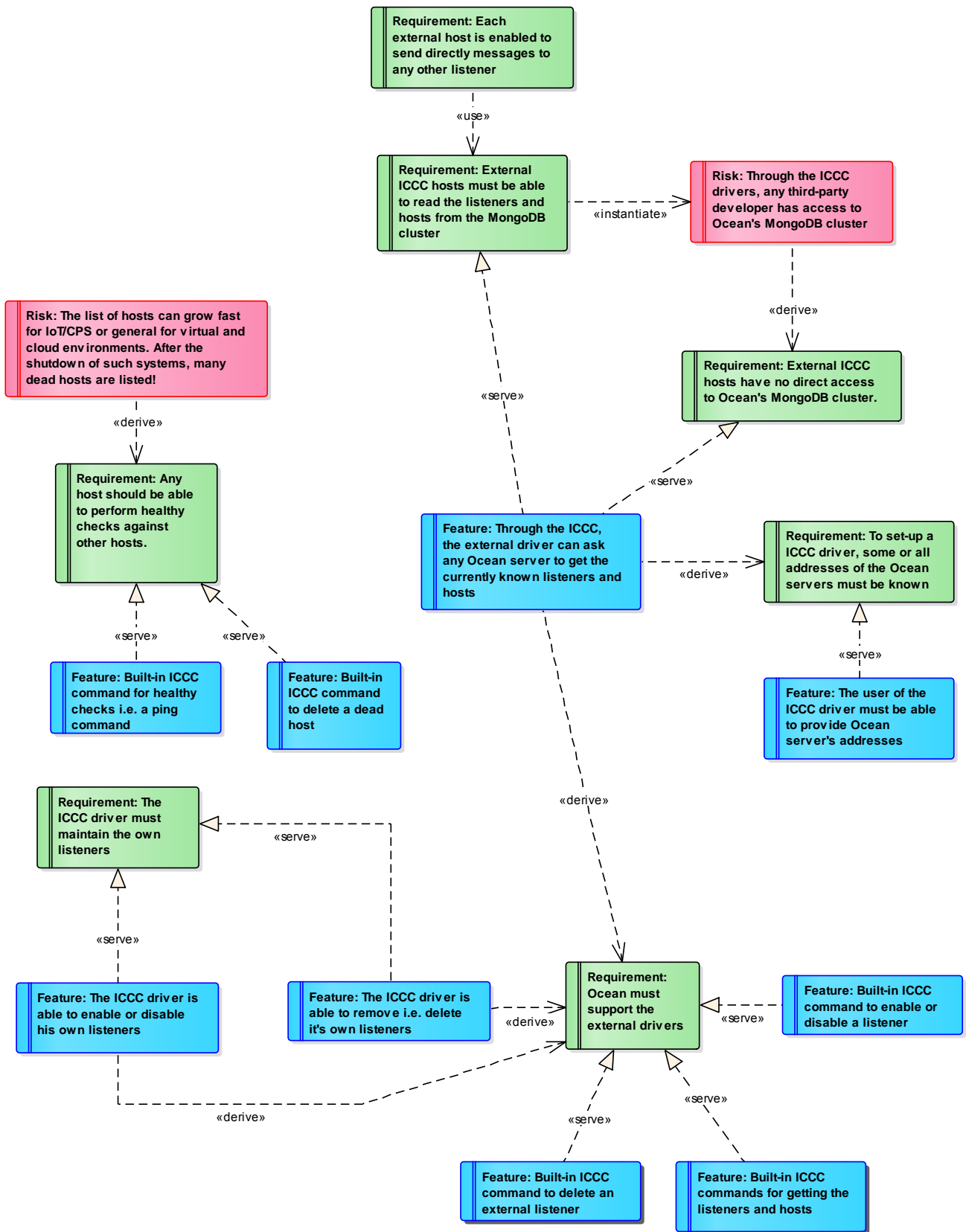
# Ocean's WriteMessage2All() function



# Ocean's WriteMessage2Any() function



# Requirements for ICCC drivers



# Class diagram for an Ocean ICC driver

ICCC
+ HandlerIncomings
+ ICCConnection
+ ICCHost
+ ICCListener
+ ICCProcessor
+ Kind
+ Registrar
+ SystemChannels
+ <i>ICCCMessage</i>
+ SystemMessages

SystemMessages
+ ICCComponentStartUpMessage
+ ICCDefaultAnswer
+ ICCDeleteHost
+ ICCDeleteListener
+ ICCGetHosts
+ ICCGetHostsAnswer
+ ICCGetListeners
+ ICCGetListenersAnswer
+ ICCGetVersion
+ ICCGetVersionAnswer
+ ICCListenerUpdate
+ ICCNewLogEvent
+ ICCNumGenNext
+ ICCNumGenNextAnswer
+ ICCOceanStartUpMessage
+ ICCPing
+ ICCRegisterHost
+ ICCRegisterListener

(from ICC)

ICCC::HandlerIncomings
+ ProcessRequest(WebRequest, WebResponse): void
«constructor»
+ HandlerIncomings()

ICCC::SystemChannels
+ ICC: 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: ICCProcessor {readOnly}
- maintainerThread: Thread
~ convertData2Message(NameValueCollection, IICCCMessage): IICCCMessage
~ convertMessage2Data(IICCCMessage): NameValueCollection
- maintainer(): void
+ setConfiguration(string): void
~ signMessage(NameValueCollection): NameValueCollection
«constructor»
- ICCProcessor()

ICCC::Kind
+ ALL: byte = 255 {readOnly}
+ Component: byte = 1 {readOnly}
+ Ocean: byte = 0 {readOnly}

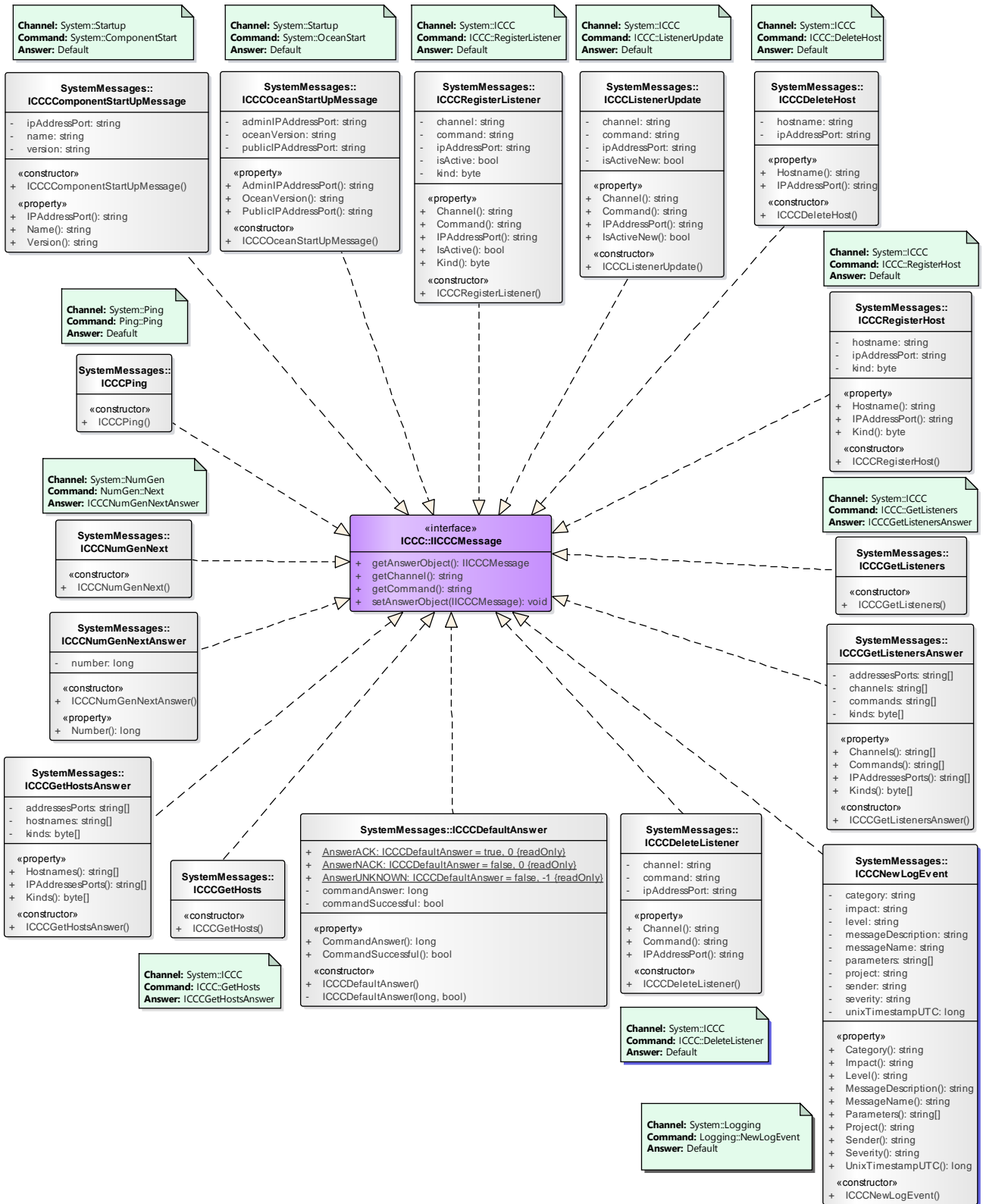
ICCC::Registrar
+ INSTANCE: Registrar {readOnly}
- localListeners: Dictionary<string, MethodCallback>
+ addLocalListener(string, string, MethodCallback): void
- getLocalListeners(string, string): IEnumerable<MethodCallback>
«constructor»
- Registrar()

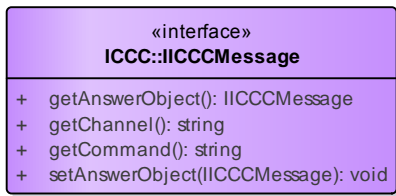
ICCCHost and ICCListener are classes to maintain lists with the current available entries.

ICCC::ICCCHost
- hostname: string
- ipAddressPort: string
- kind: byte
«property»
+ Hostname(): string
+ IPAddressPort(): string
+ Kind(): byte
«constructor»
+ ICCHost()
+ ICCHost(string, string)

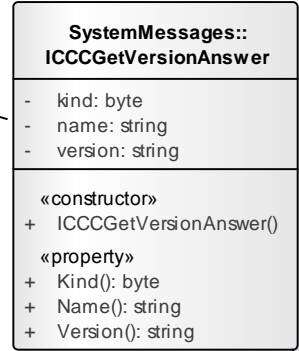
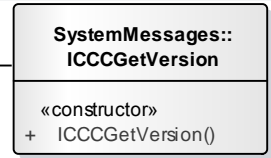
ICCC::ICCCConnection
- icccHostsCache: IEnumerable<ICCCHost>
- icccListenerCache: IEnumerable<ICCCListener>
+ INSTANCE: ICCConnection {readOnly}
+ send2All(IICCCMessage, byte): IEnumerable<IICCCMessage>
+ send2Any(IICCCMessage, byte): IICCCMessage
~ sendMessage(NameValueCollection): void
+ setConfiguration(IEnumerable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

ICCC::ICCCListener
- channel: string
- command: string
- ipAddressPort: string
- isActive: bool
- kind: byte
«property»
+ Channel(): string
+ Command(): string
+ IPAddressPort(): string
+ IsActive(): bool
+ Kind(): byte
«constructor»
+ ICCListener()
+ ICCListener(string, string, string, bool)



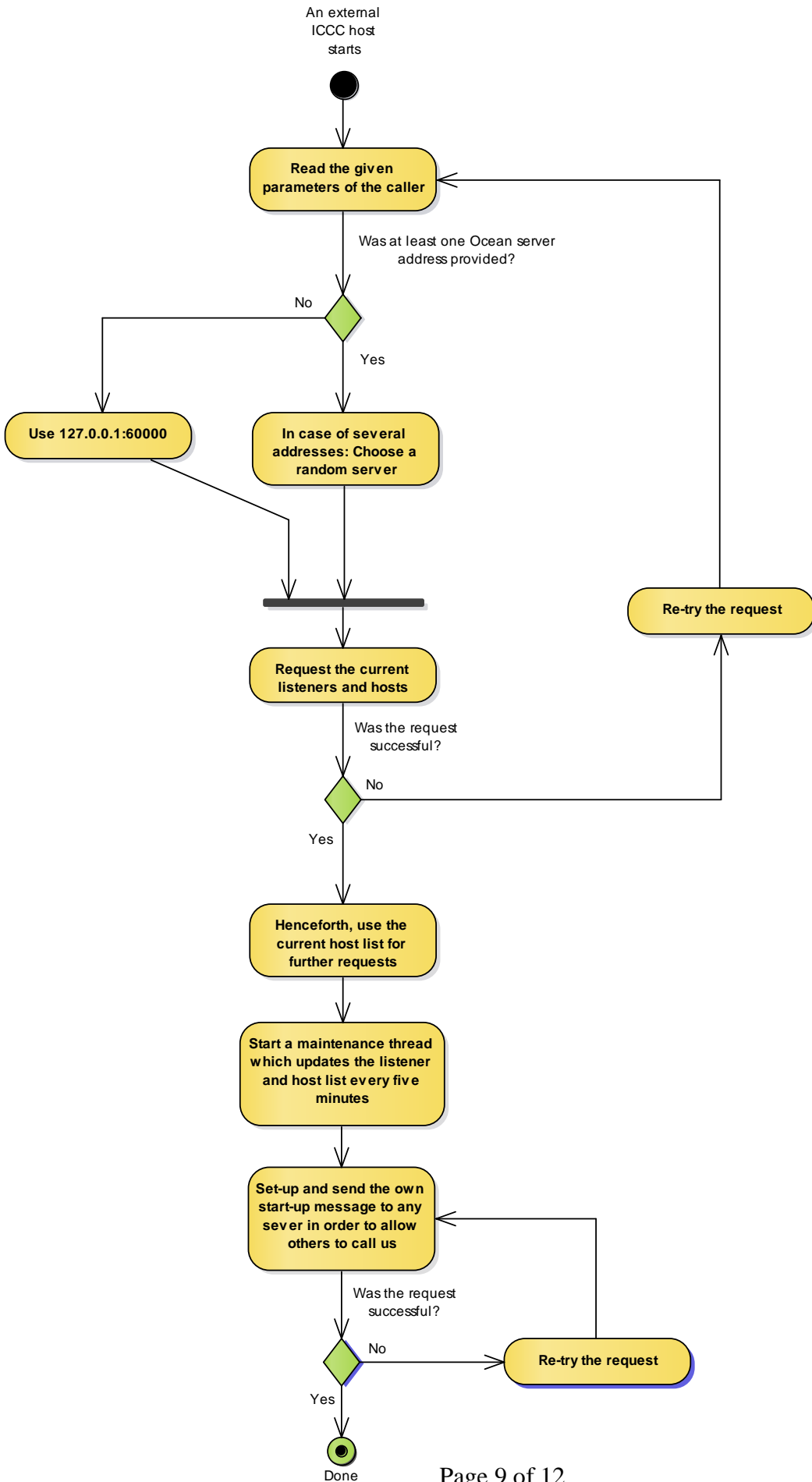


**Channel:** System  
**Command:** System::Version  
**Answer:** ICCCGetVersionAnswer





# Start-up process for a external ICCC driver / host



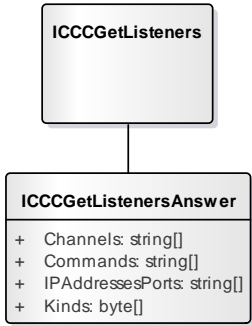
# Documentation of ICCC Messages

**Message: ICCCGetListeners**

**Channel:** System::ICCC  
**Command:** ICCC::GetListeners  
**Answer:** ICCCGetListenersAnswer

**Behavior Ocean internal:**  
 ... Send: Never, will use internal function  
 ... Receive: Provides all known listeners

**Behavior external components:**  
 ... Send: To obtain all listeners  
 ... Receive: Never

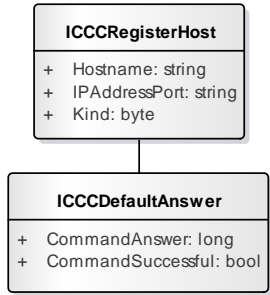


**Message: ICCCRestHost**

**Channel:** System::ICCC  
**Command:** ICCC::RegisterHost  
**Answer:** Default

**Behavior Ocean internal:**  
 ... Send: Never, will use internal function  
 ... Receive: Registers the host

**Behavior external components:**  
 ... Send: To register own host  
 ... Receive: Never

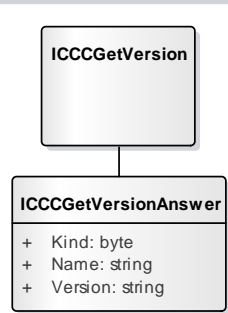


**Message: ICCCGetVersion**

**Channel:** System  
**Command:** System::Version  
**Answer:** ICCCGetVersionAnswer

**Behavior Ocean internal:**  
 ... Send: On admin's request  
 ... Receive: Reports own version; Kind=Ocean; Name=Hostname

**Behavior external components:**  
 ... Send: On admin's request  
 ... Receive: Reports own version; Kind=Component; Name=Component's name

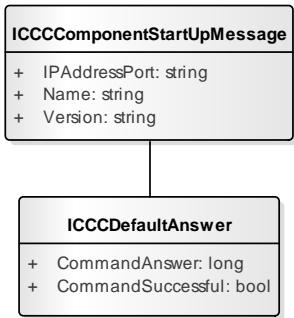


**Message: ICCComponentStartUpMessage**

**Channel:** System::Startup  
**Command:** System::ComponentStart  
**Answer:** Default

**Behavior Ocean internal:**  
 ... Send: Never  
 ... Receive: Log the content

**Behavior external components:**  
 ... Send: After start-up  
 ... Receive: Logging is possible

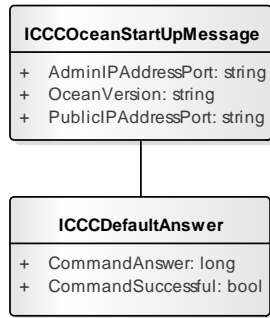


**Message: ICCOceanStartUpMessage**

**Channel:** System::Startup  
**Command:** System::OceanStart  
**Answer:** Default

**Behavior Ocean internal:**  
 ... Send: After start-up of any Ocean server  
 ... Receive: Log the content

**Behavior external components:**  
 ... Send: Never  
 ... Receive: Logging is possible

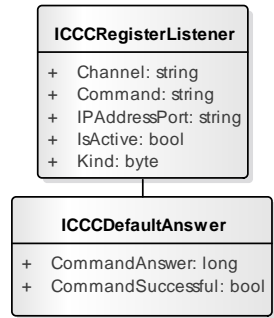


**Message: ICCRegisterListener**

**Channel:** System::ICCC  
**Command:** ICCC::RegisterListener  
**Answer:** Default

**Behavior Ocean internal:**  
 ... Send: Never, will use internal function  
 ... Receive: Register the listener

**Behavior external components:**  
 ... Send: Anytime to register own listeners.  
 ... Receive: Never

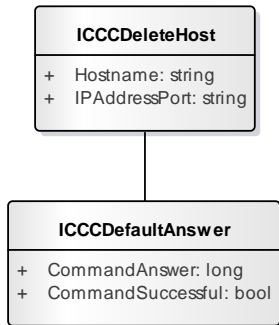


**Message: ICCDeleteHost**

**Channel:** System::ICCC  
**Command:** ICC::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

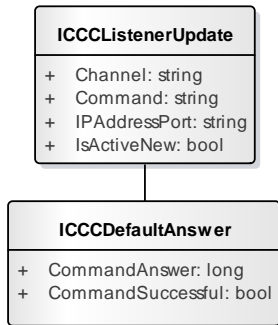


**Message: ICCListenerUpdate**

**Channel:** System::ICCC  
**Command:** ICC::ListenerUpdate  
**Answer:** Default

**Behavior Ocean internal:**  
... Send: Never  
... Receive: Update the IsActive status for this listener

**Behavior external components:**  
... Send: To update known listeners  
... Receive: Never

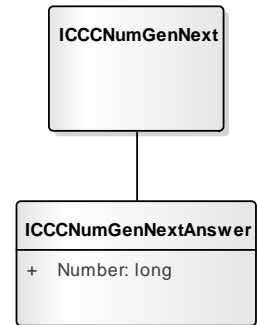


**Message: ICCNumGenNext**

**Channel:** System::NumGen  
**Command:** NumGen::Next  
**Answer:** ICCNumGenNextAnswer

**Behavior Ocean internal:**  
... Send: Never, will use internal function  
... Receive: Delivers a new unique number

**Behavior external components:**  
... Send: To obtain a unique number  
... Receive: Never

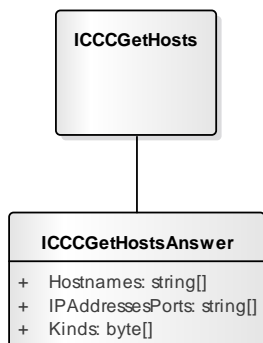


**Message: ICCGetHosts**

**Channel:** System::ICCC  
**Command:** ICC::GetHosts  
**Answer:** ICCGetHostsAnswer

**Behavior Ocean internal:**  
... Send: Never, will use internal function  
... Receive: Delivers current known hosts

**Behavior external components:**  
... Send: To obtain all known hosts  
... Receive: Never

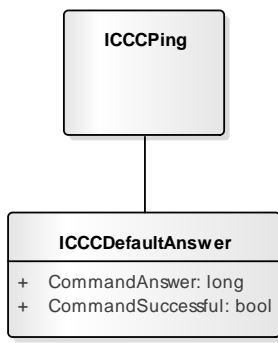


**Message: ICCPing**

**Channel:** System::Ping  
**Command:** Ping::Ping  
**Answer:** Default

**Behavior Ocean internal:**  
... Send: To check the status of other hosts  
... Receive: Assume the host as healthy

**Behavior external components:**  
... Send: To check the status of other hosts  
... Receive: Assume the host as healthy

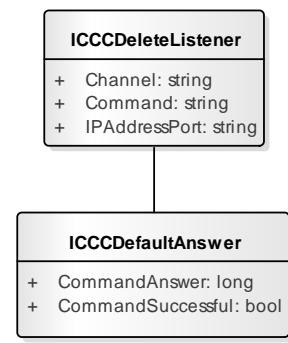


**Message: ICCDeleteListener**

**Channel:** System::ICCC  
**Command:** ICC::DeleteListener  
**Answer:** Default

**Behavior Ocean internal:**  
... Send: Never, will use internal function  
... Receive: Deletes the listener

**Behavior external components:**  
... Send: To maintain listeners  
... Receive: Never



**Message: ICCCNLogEvent**

**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

