CFS Application - SBN
Software Bus Network Services
IP Version

Current Status Report
February 28, 2007

Robert McGraw

Code 582/SSI

rmcgraw@hst.nasa.gov

301-286-5069

SBN - What We Currently Have

- CFS Application for inter-processor communication (SBN no longer embedded in the cFE)
- IP Version Only
- Designed to run on Ethernet, Spacewire, SOIS and 1394 (Firewire) without SBN code changes
- Learns of Peers via Peer Configuration File
- Can be used with cFE Version 4.1.0 or later
- Requires no changes to existing cFE/CFS applications

SBN - What We Don't Have

- Quality Of Service not yet in use (but hooks are in)
- Cannot Interface to 1553 - may need a separate 1553 version of SBN
- There is no way to send large pkts to peers. Currently maximum network packet size is constrained by MTU (usually ~1500 bytes)
- Cannot Learn of Peers that are not listed in the Peer Configuration File
- Cannot Send Raw Packets (Non IP) to peers
- Bridging two subnets is not yet supported (some hooks are in).

SBN - Deviation from September Plan

At the September 2006 meeting, we decided to abstract SBN so that the same SBN code can be used with or without IP on Ethernet, Spacewire, 1553, or SOIS. This was to be done by having generic functions for the following:

- to initialize an interface (ifinit)
- to send a message to any destination (ifsend)
- to receive a message from any destination. (ifrcv)
- to terminate an interface (ifclose)

This idea is possible, but to try and cover all possibilities with the same SBN would add significant complexity to an already complex application.

It seems to be more feasible to have an IP version of SBN that is separate from the non IP version and possibly a third version that interfaces to 1553.

SBN - Configuring the Nodes

To configure each node the user must:

- Create the Peer Configuration File:

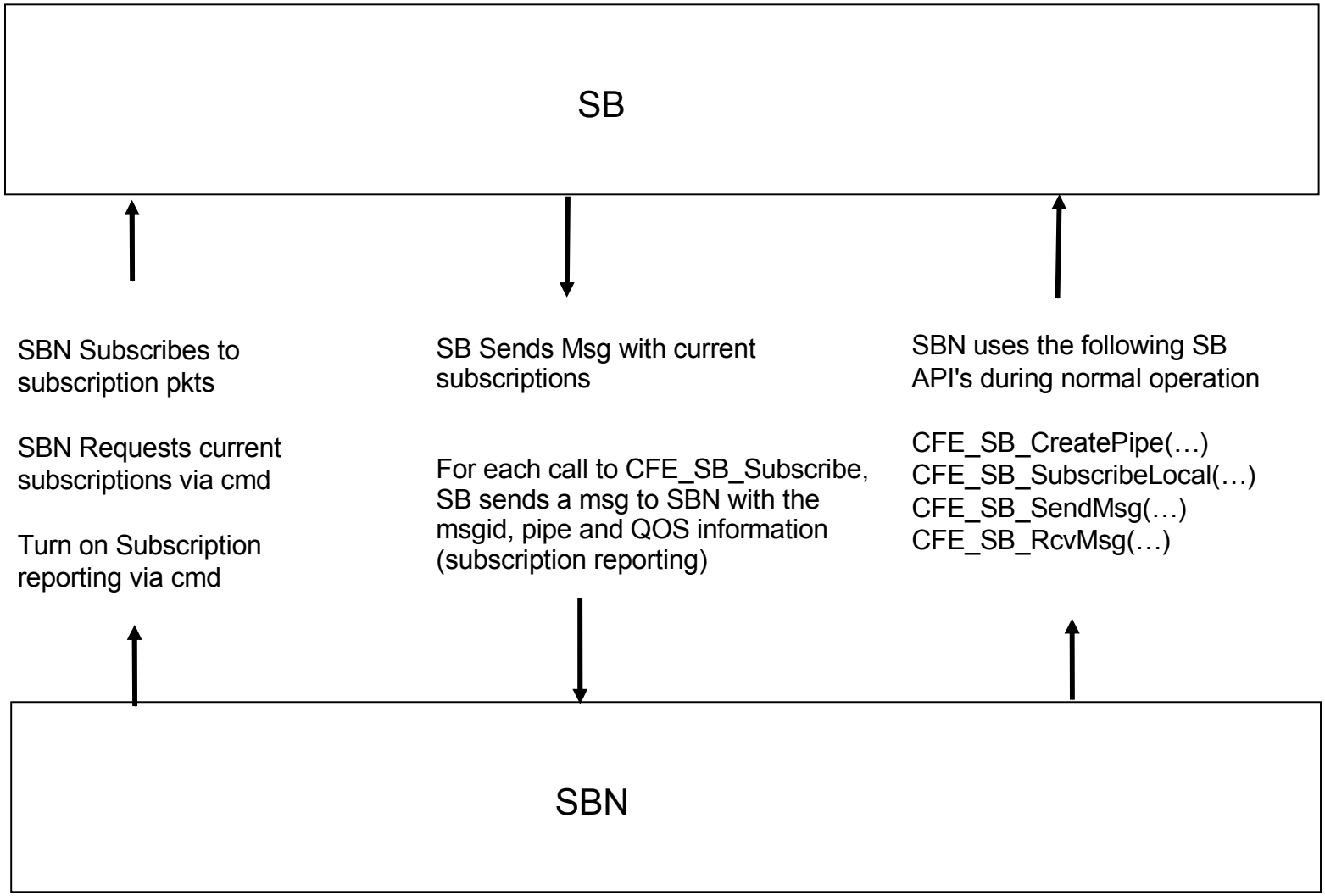
  Define Peer Name, IP Address, Data Port and Protocol Port for each peer.
  CPU1, 192.168.001.004, 15820, 5820;
  CPU2, 192.168.001.006, 15820, 5821;
  CPU3, 192.168.001.008, 15820, 5822;

  Each node gets a copy of the same Peer Configuration File

- Set values for the following prior to building (if default is insufficient)

  SBN_MAIN_LOOP_DELAY, default is 1 second,
  SBN_TIMEOUT_CYCLES, default is set to 5
  SBN_VOL_PEER_FILENAME, default "/ram/SbnPeerData.dat"
  SBN_NONVOL_PEER_FILENAME, default "/cf/SbnPeerData.dat"
  SBN_MAX_SUBS_PER_PEER, default 256

SBN - SB Interface



SBN - Peer Table

The subscription list in the peer table has two purposes:

1. allows the routing to be cleaned up whenever a peers heartbeat stops
2. provides the Qos to the SBN sending the message over the network

Peer 0

Peer Name	"CPU2"		
Peer State	Heartbeating		
Dest Adr	192.168.1.6		
Pipe ID	10		
Heartbeat Timer	3		
Subscription Count	3		
Subscription 0	MsgId 0x1842	Reliability 0	Priority 0
Subscription 1	0x1801	1	0
Subscription 2	0x081A	1	1
	○ ○ ○		
Subscription 256			

Peer 1

Peer Name	"CPU3"		
Peer State	Heartbeating		
Dest Adr	192.168.1.8		
Pipe ID	12		
Heartbeat Timer	1		
Subscription Count	2		
Subscription 0	MsgId 0x0810	Reliability 0	Priority 0
Subscription 1	0x1865	0	0
Subscription 2			
	○ ○ ○		
Subscription 256			

Peer tables for the "cpu1" processor

SBN - Network Message Types

SBN uses 7 unique network messages :

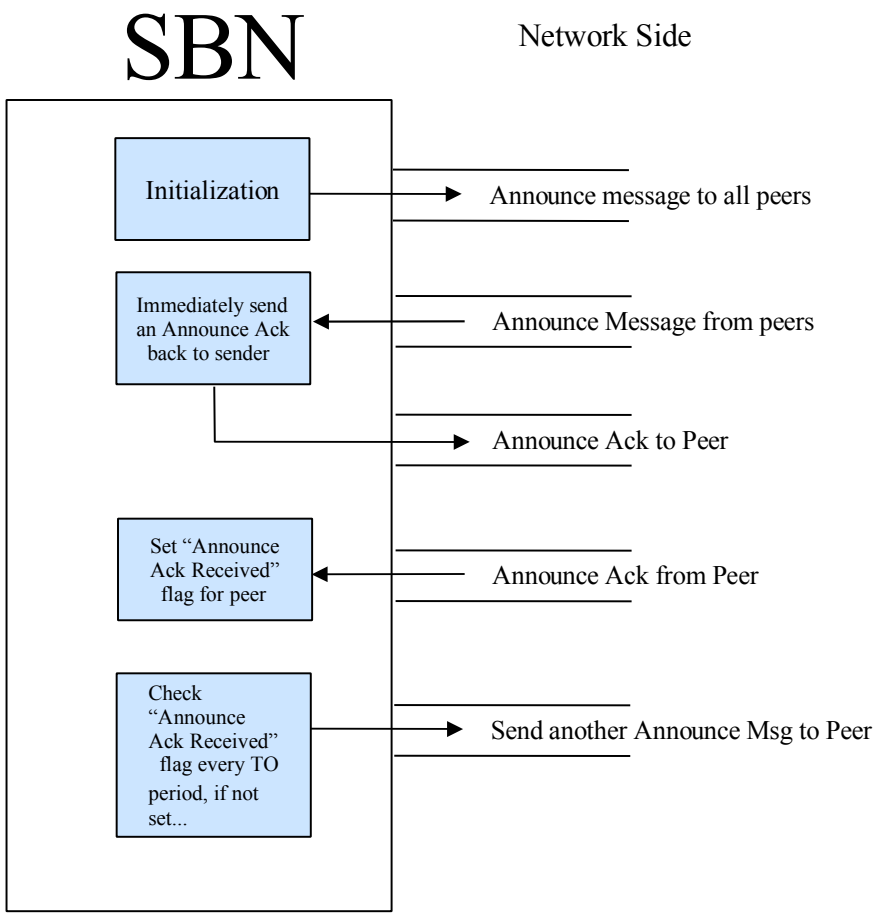
- ANNOUNCE - Sent on initialization to all peers.
- ANNOUNCE ACK - Sent in response to an ANNOUNCE
- HEARTBEAT - Sent periodically to all peers
- HEARTBEAT ACK - Sent in response to HEARTBEAT
- SUBSCRIBE - Sent to all peers when a task subscribes to a message (via CFE_SB_Subscribe)
- UNSUBSCRIBE - Sent to all peers when a task unsubscribes to a message (via CFE_SB_Unsubscribe)
- APP MESSAGE - Sent to all subscribing peers when a message is read from a pipe

SBN - Network Message Formats

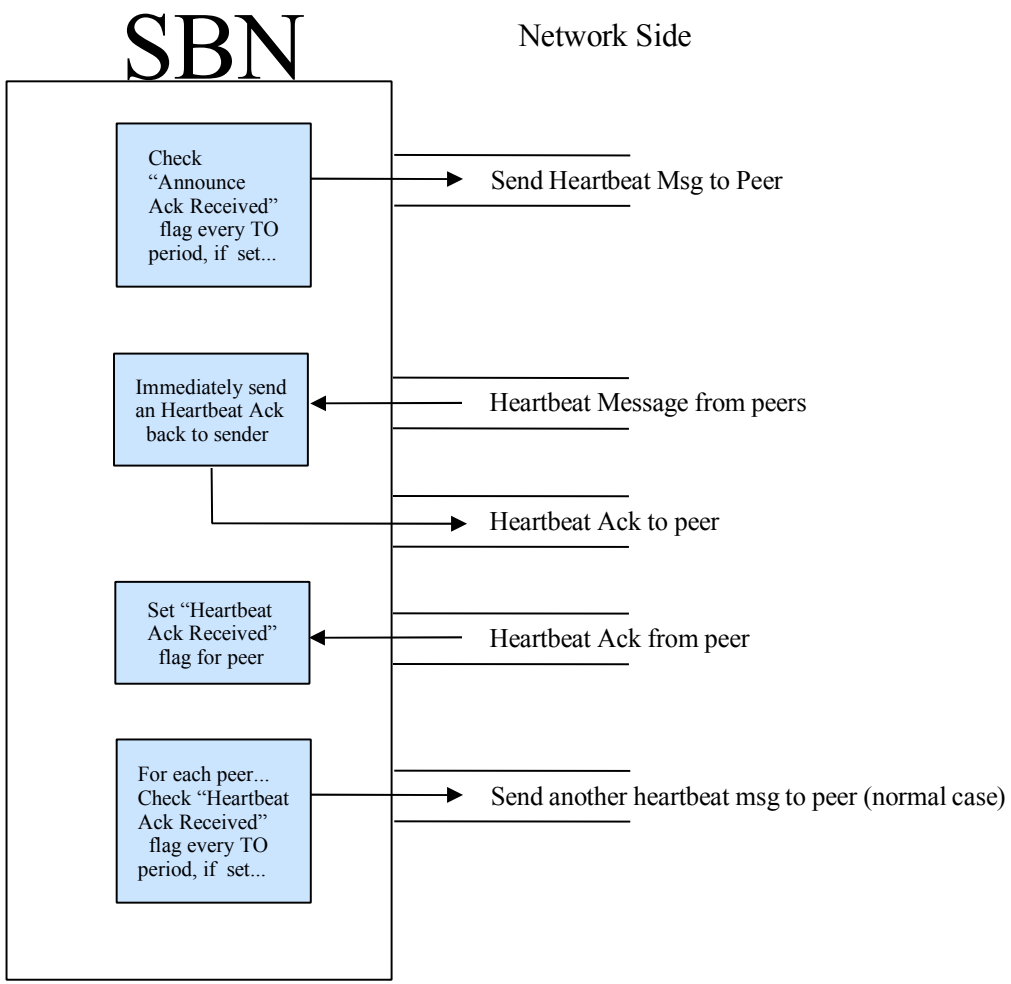
SBN appends a custom header to each network message so that the message type and the sender are easily identified by the receiving SBN

	SBN HDR		SBN DATA PORTION
	uint32 Msg Type	8- byte string	Number of bits are shown in parenthesis
ANNOUNCE	0x10	SENDER	UNUSED
ANNOUNCE ACK	0x11	SENDER	UNUSED
HEARTBEAT	0x20	SENDER	UNUSED
HEARTBEAT ACK	0x21	SENDER	UNUSED
SUBSCRIBE	0x30	SENDER	MsgId (16) | QOS - Reliability (8) | QOS - Priority (8)
UNSUBSCRIBE	0x40	SENDER	MsgId (16) | QOS - Reliability (8) | QOS - Priority (8)
APP MSG	0x50	SENDER	Variable Length

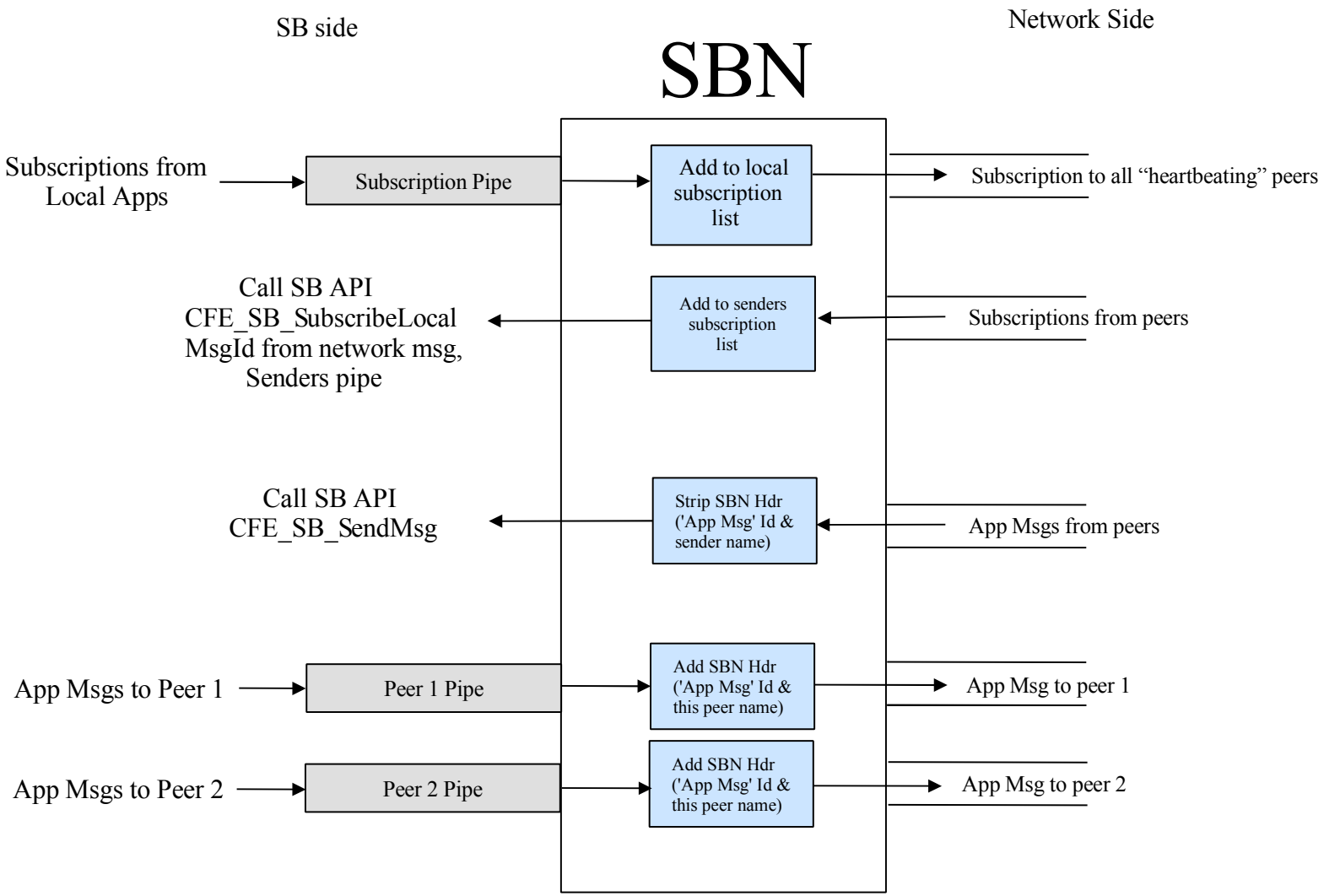
SBN – Announce Message Processing



SBN – Heartbeat Message Processing



SBN – Data Port Message Processing



SBN – PDL Page 1

```
SBN Task Main

Initialization
    Create Subscription Pipe
    Send Cmd to SB to turn on subscription reporting
    Request a list of previous subscriptions from SB
    Create Cmd Pipe

    Loop Through Peers listed in  configuration file
        if (entry == self)
            Create protocol socket
            bind protocol socket to prtotocol port listed in file
            Create data socket
            bind data socket to data port listed in file
        else
            Create Pipe for peer. (Enter Pipe Id in Channel Tbl)
            Create Socket for peer. (Enter socket Id in Channel Tbl)
            Send ANNOUNCE Msg to peer
    End Loop Through Entries in configuration file
End Initialization

Forever
    Wait 1sec (wait time is a configurable parameter)
    Check for Incoming Network Messages (see next page for more detail)
    RunProtocol (see pdl page 3 for more detail)
    Check Subscription Pipe (see pdl page 4 for more detail)
    Check Peer Pipes (see pdl page 4 for more detail)
    Check Cmd Pipe (see pdl page 4 for more detail)
End Forever

End SBN Task Main
```

SBN – PDL Page 2

```
SBN Task Main
Initialization

Forever
    Wait 1sec (wait time is a configurable parameter)
    Check for Incoming Network Messages
        If ANNOUNCE,  find sender, send ANNOUNCE ACK back
            If ANNOUNCE ACK, find sender, set “announce ack rcvd flag” for sender
            If HEARTBEAT, find sender, send HEARTBEAT ACK back
            If HEARTBEAT ACK, find sender, set “heartbeat ack rcvd flag” for sender
            If SUBSCRIBE,
                find sender
                log subscription in peer's subscription list
                call CFE_SB_SubscribeLocal,  give MsgId from network msg and PipeId from peer table,

            If UNSUBSCRIBE,
                find sender
                remove subscription from peer's subscription list
                call CFE_SB_UnsubscribeLocal,  give MsgId from network msg and PipeId from peer table,

            If APP MSG, strip SBN Hdr (APP MSG identifier) and send message to SB for routing (via CFE_SB_SendMsg)
    End Check for Incoming Network Messages

    RunProtocol
    Check Subscription Pipe
Check Peer Pipes for SB Message
    Check Command Pipe for Message

End Forever
End SBN Task Main
```

SBN – PDL Page 3

```
SBN Task Main
Initialization
    Forever
        Wait 1sec (wait time is a configurable parameter)
        Check for Incoming Network Messages
        RunProtocol
            Loop Through Peers
                peer timer++
                if peer timer >= timeout_cycles (cfg param, default = 5)
                    reset timer
                    if peer state == ANNOUNCING
                            if “announce ack rcvd flag” is set
                                    change peer state to HEARTBEATING
                                    send local subscriptions to peer
                                    send heartbeat message to peer
                            else if “announce ack rcvd flag” not set
                                    send another announce message to peer
                            end if announce ack rcvd flag...
                    else if peer state == HEARTBEATING
                            if “heartbeat ack rcvd flag” is set
                                    send another heartbeat message to peer
                            else if “heartbeat ack rcvd flag” is not set
                                    change state back to ANNOUNCING
                                    send “heartbeat lost” event
                                    unsubscribe to all subscriptions from peer
                                    send announce message to peer
                            end if heartbeat ack rcvd flag..
                    end if peer state ....
                end if peer timer...
            end loop through peers
        End RunProtocol
        Check Subscription Pipe
        Check Peer Pipes for SB Message
        Check Command Pipe for Message
    End Forever
End SBN Task Main
```

SBN – PDL Page 4

```
SBN Task Main
Initialization

Forever
    Wait 1sec (wait time is a configurable parameter)
    Check for Incoming Network Messages
    RunProtocol

    Check Subscription Pipe (for local subscribes and unsubscribes)
        If Subscription found
            Log subscription in local subscription list
            Add SUBSCRIPTION identifier and send to all peers w/ state = HEARTBEATING
        If Unsubscription found
            Remove subscription from local subscription list
            Add UNSUBSCRIPTION identifier and send to all peers w/ state = HEARTBEATING
    end Check Subscription Pipe

    Check Peer Pipes for SB Message
        Loop Through Peers w/ state = HEARTBEATING
            If Message found
                Add APP MSG identifier and send to proper network peer
end Check Peer Pipes

    Check Command Pipe for Message
        If Request for HK, Send HK Msg
        If Ground Command, Process Ground Command
    end Check Command Pipe

End Forever
End SBN Task Main
```

SBN - Resets

- **Power-On and Processor Resets:**
  - All connections with peers are terminated.
  - All subscriptions are lost and need to be recreated during initialization.

SBN – Commands

Command	Function Code	Parameters	Description
No-op	0x1	None	Increments command execution counter
Reset Counters	0x2	None	Reset command execution counter and command error counters to zero
Send All Peer Info	0x3	None	Writes all ram peer information to telemetry file
Send Single Peer Info	0x4	peer name	Sends Single Peer info in event
Set forever loop wait time	0x5	wait time in mS	Sets wait time in milliseconds
Set heartbeat timeout cycles	0x6	Timeout cycles	Sets heartbeat timeout in # of forever loop cycles
Send Network Diagnostic pkt	0x7	None	Sends network diagnostic information in packet

SBN - Telemetry

Command Counter
Command Error Counter

Names of Peers
State of Peers

Count of pkts sent to each peer
Count of pkts recvd from each peer

Count of local subscriptions
Count of subscriptions received by each peer

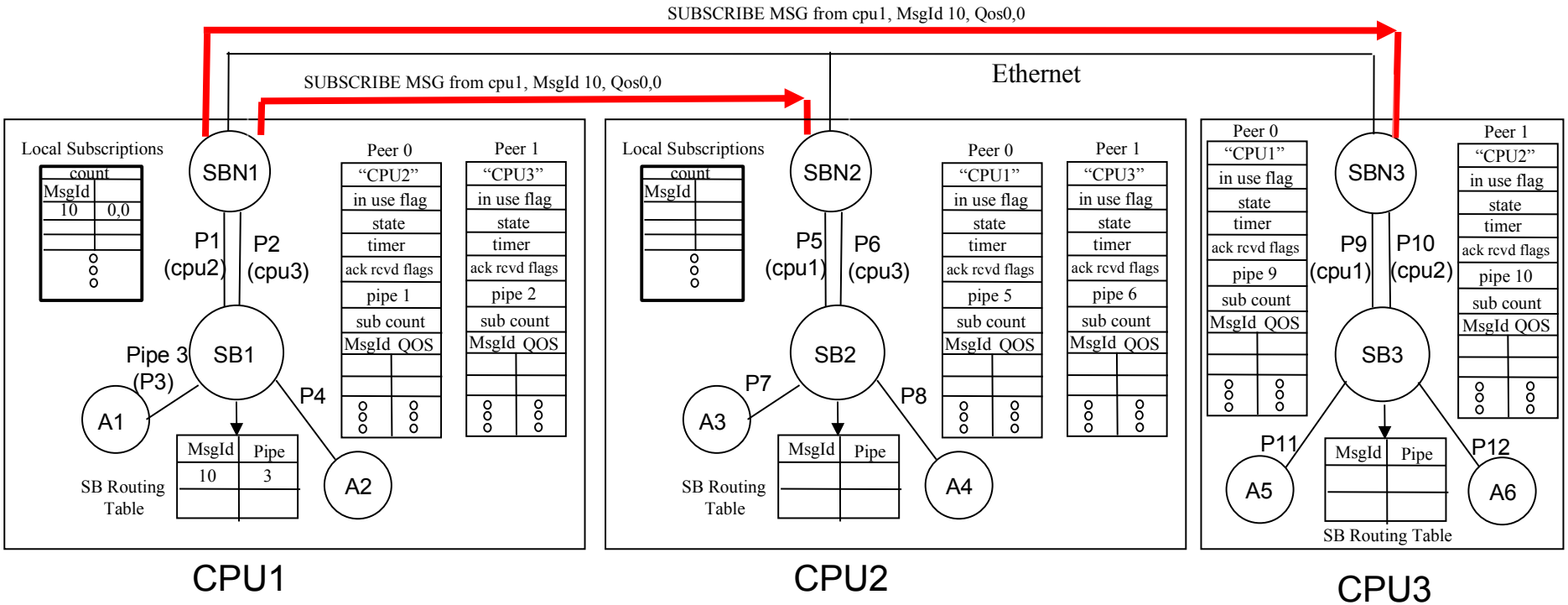
SBN – Error Event Messages

- **Error Event Messages**
  - SBN APP Will Terminate, Peer File Not Found or Data Invalid!
  - SBN APP Will Terminate, Error Creating Interfaces!
  - SBN APP Initialized V1.0, AppId=%d
  - "%s:Peer file %s failed to open", CFE_CPU_NAME,Nonvolatile Filename
  - "%s:Invalid SBN peer file line,exp %d items,found %d"
  - "%s:Error copying file data for %s,status=0x%x"
  - "%s:Error creating pipe for %s,status=0x%x"
  - "%s:socket call failed,line %d,rtn val %d,errno=%d"
  - "%s:bind call failed,line %d,rtn val %d,errno=%d"
  - "%s:Unexpected state(%d) in SBN_RunProtocol for %s"
  - "%s:Error recving network message, srcName %s invalid"
  - "%s:Cannot process subscription from %s,max(%d)met."
  - "%s:Error sending %s to %s stat=%d"

SBN – Informational Event Messages

- **Informational Event Messages**
  - “SBN APP Initialized V1.0, AppId=%d”
  - "%s:Peer file %s failed to open", CFE_CPU_NAME,Volatile Filename
  - "%s:Opened SBN Peer Data file %s"
  - "%s:Pipe %s created"
  - "%s:%s Alive, changing state to SBN_HEARTBEATING"
  - "%s:%s Heartbeat lost, changing state to SBN_ANNOUNCING"
  - "%s:AppMsg 0x%04X,sz=%d destined for %s truncated to %d(max sz)"
  - "%s:Error sending subs to %s,LclSubCnt=%d,max=%d"
  - "%s:UnSubscribed %d MsgIds from %s"

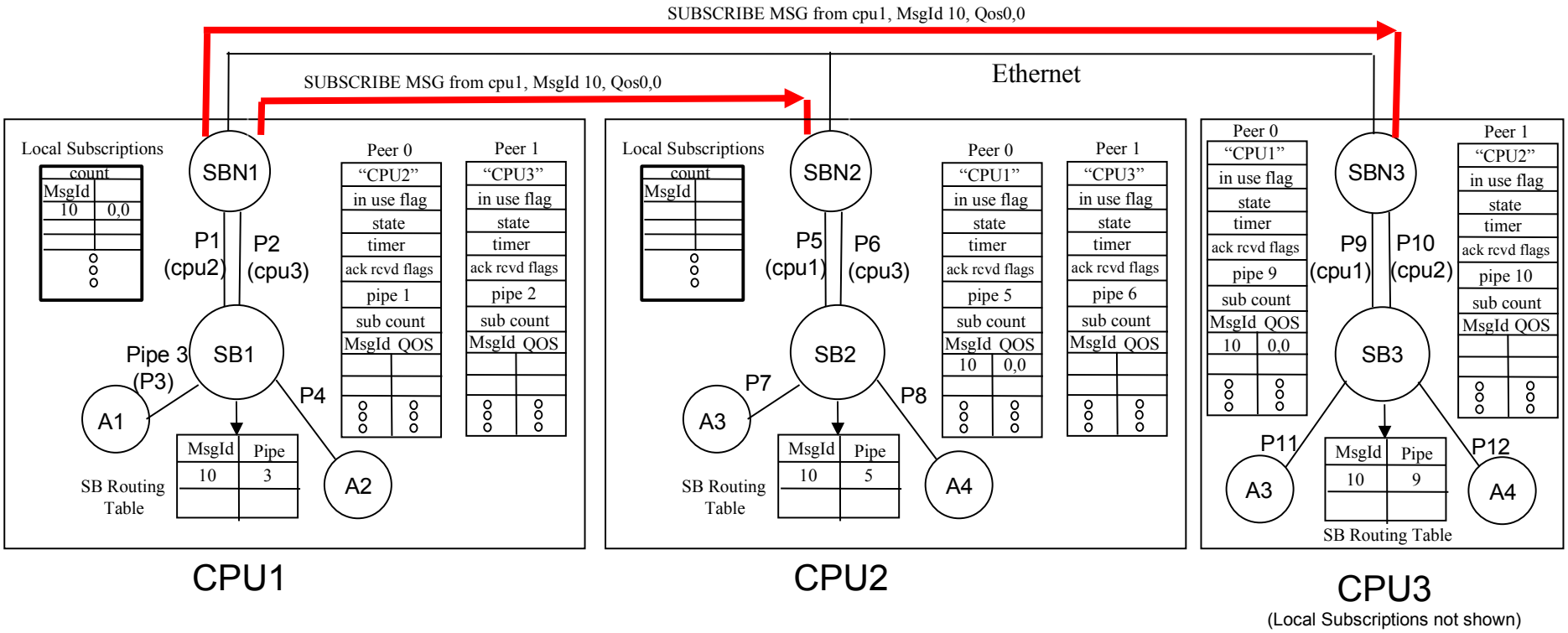
SBN – Subscription Example (1 of 2)



In the following slides, P#'s refer to pipes and A#'s refer to applications.

1. A1 subscribes to MsgId 10 on Pipe 3 (P3) with QOS of 0,0 (meaning low reliability, low priority).
2. SB1 does the following:
   a. Updates 'CPU1' SB routing tables with MsgId 10 to Pipe 3.
   b. Sends a message to SBN1 with subscription details (subscription report)
3. SBN1 does the following after receiving the subscription report:
   a. Adds subscription to "Local Subscription" list
   b. Sends a "SUBSCRIBE" network message to all "heartbeating" peers

SBN – Subscription Example (2 of 2)

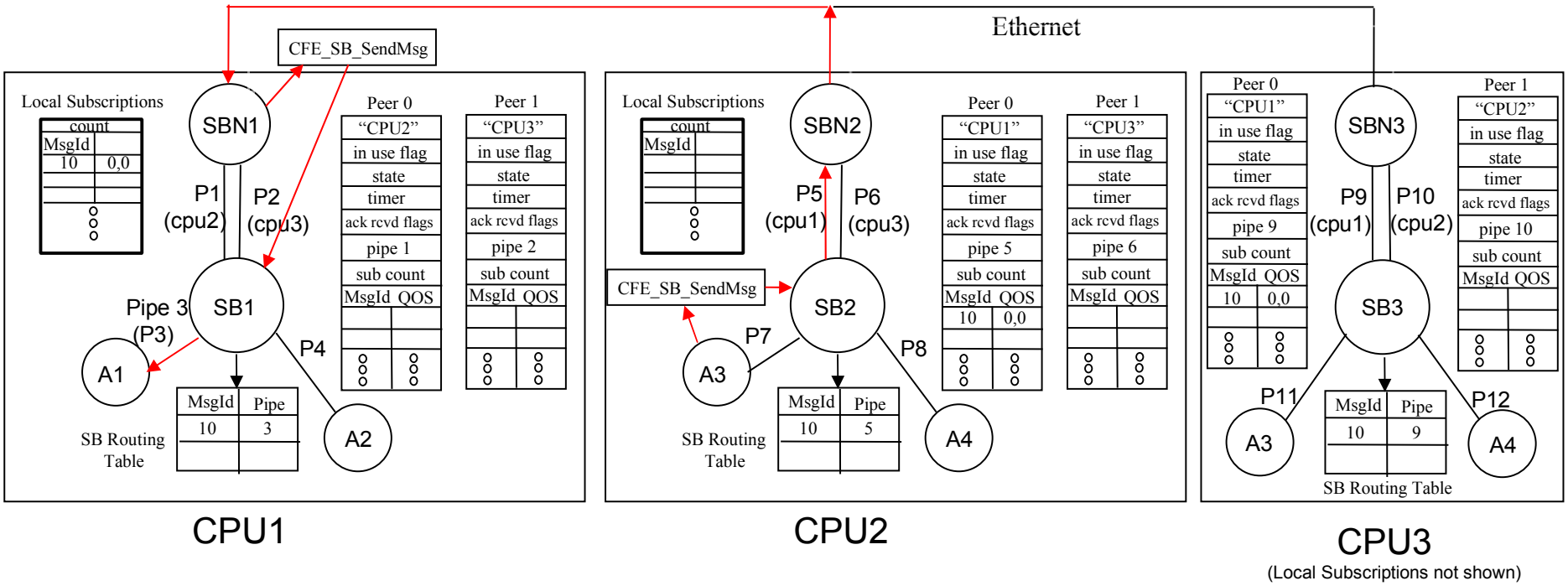


In response to the network 'SUBSCRIBE' message, SBN2 will :

- Find the sender of the message (cpu1)
- Look up the pipe number for the sender in the channel table (5)
- Do a CFE_SB_SubscribeLocal of msgId 10, on cpu1 pipe.(5)
- Add the MsgId and QOS to its subscription list for CPU1.

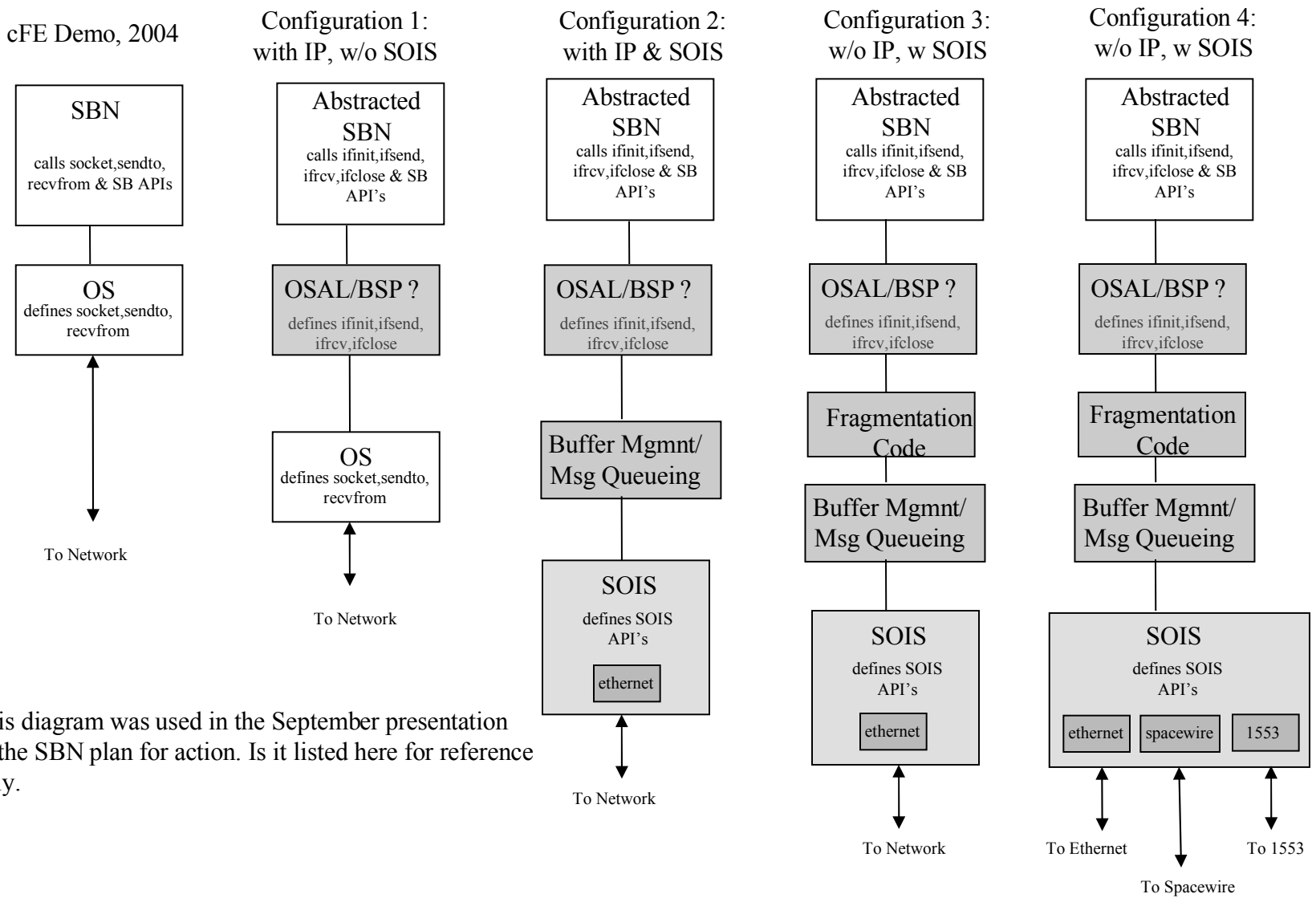
SBN3 will respond to the SUBSCRIBE message in the same way.

SBN – Message Send Example



Application A3 sends a message with message ID 10.
SB routes the message to pipe 5.
SBN2 reads the message off pipe 5, adds the SBN Hdr (APP MSG identifier and Src Name=CPU2), then sends it to CPU1. (All pipe 5 pkts go to CPU1)
SBN1 reads the message off the network, strips the APP MSG identifier and sends it to SB for routing.
Inside the CFE_SB_SendMsg API, the message is routed to pipe 3.
Application A1 receives the message via the CFE_SB_RcvMsg API.

SBN – for reference



This diagram was used in the September presentation of the SBN plan for action. Is it listed here for reference only.