

**Component Definition Document
(CDD)
For the
External Adapter
Example
Component Assembly**

Rev. -

July 12, 2011

***Prepared By:*
Northrop Grumman Corporation
Electronic Systems
Baltimore, MD**

Table of Contents

1 Introduction.....	3
1.1 Scope.....	3
2 Applicable Documents.....	3
2.1 Applicable Government Documents.....	3
2.2 Other Applicable Documents.....	3
3 Component Description.....	3
3.1 Overview.....	3
3.2 Operational Context.....	4
4 Component Interfaces.....	4
4.1 Service Ports.....	4
4.1.1 Relay Server Facet (internal).....	4
4.2 Client Ports.....	5
4.2.1 Relay Server Receptacle (internal).....	5
4.3 Publisher Ports.....	5
4.4 Subscriber Ports.....	5
4.5 External Ports.....	5
5 Component Functionality.....	5
6 Configurable Parameters.....	5
7 Design Constraints.....	6
8 Component Test.....	6
9 Component Dependencies.....	6

1 Introduction

1.1 Scope

This document captures the specification and design for the External Adapter Example software component assembly. This example assembly is targeted for deployment on the Scalable Node Architecture (SNA) real-time component framework. As such, it must be compliant with SNA Component Based Architecture (CBA) design guidelines.

This specification defines the component assembly's functional, interface and performance requirements, the context in which it must operate, and any design constraints it must adhere to. It provides criteria for verifying compliance, but it does not state methods for achieving results.

This is intended to be a relatively informal living document, to be included in same CM repository and package as the component source code. This CDD will initially be populated by a system engineer or software architect/lead to define component design constraints & guidelines. Over time, it will transition to enhance the "to be built" specification sections with "as built" design information documenting the final component product.

2 Applicable Documents

2.1 Applicable Government Documents

Document No.	Title

2.2 Other Applicable Documents

Document No.	Title

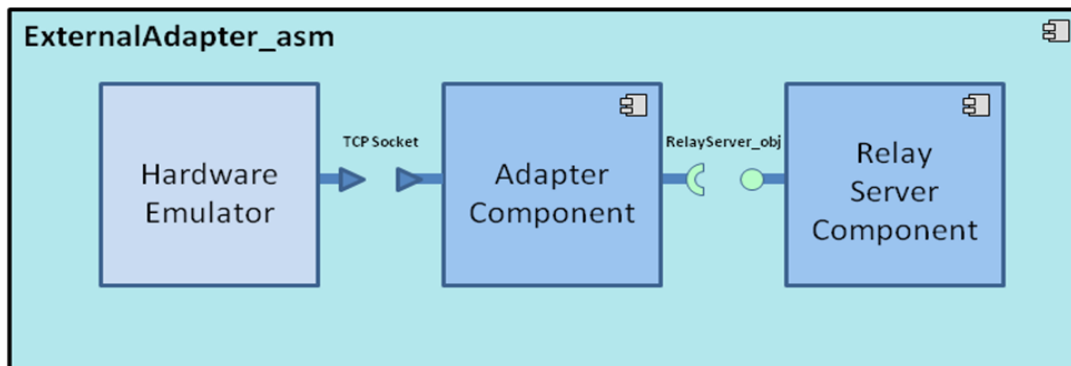
3 Component Description

3.1 Overview

The External Adapter Example component assembly is one of the component source examples included in the SNA SDK for reference, testing and experimentation. It illustrates the pattern of using an adapter component to translate an external event into a CORBA interface. In this particular example, there is a binary executable called the "Hardware Emulator". The "Hardware Emulator" will periodically generate TCP socket text message to be received by one of the Adapter Component's TCP sockets. In its current form, the Adapter Component utilizes a TCP Client class, to encapsulate the functionality of connecting to a Hardware Emulator's TCP socket. The TCP Client then has an Event Handler, which processes the socket and translates the stream to use a CCM port provided by the Relay Server component. The Relay Server will

display the contents of the text message. The Adapter Component may have multiple TCP Clients, and for each socket connection another instance of the Hardware Emulator will need to be run. In its current state, the example has the Adapter Component set up with two sockets, requiring two Hardware Emulators to be launched. The assembly containing the Hardware Emulator and two components is shown in below.

Figure 3-1 – External Adapter Example Component Assembly



3.2 Operational Context

This simple assembly is completely self contained and has no external connections. It is designed to operate solely within the constraints of the SNA SDK development environment to allow a new user/developer to step through the SNA component based development (CBD) process of loading a component assembly into the SNA IDE, building/compiling it, and then executing it.

The example is provided with an appropriate set of SNA configuration files and a deployment plan to support its execution within a single-host SNA SDK “localhost” Virtual Machine (VM). Alternative variations on the default supplied design and deployment are possible via experimentation by a software developer.

4 Component Interfaces

The component assembly has an external interface between the Hardware Emulator and the Adapter component. In addition, the External Adapter assembly defines a single internal CCM interface connection between the Adapter and Relay Server components as shown in .

4.1 Service Ports

4.1.1 Relay Server Facet (internal)

The Relay Server facet is provided by the Relay Server component. The facet interface provides one method called `echoText` which takes a string as an input argument. The Relay Server component will output the text via `log4cxx`.

4.2 Client Ports

4.2.1 Relay Server Receptacle (internal)

The Relay Server receptacle is used by the Adapter component. The receptacle interface uses one method called `echoText` which takes a string as an input argument. The Adapter component will call this method providing the text processed by a TCP socket event.

4.3 Publisher Ports

There are no internal or external publisher ports defined for this component assembly.

4.4 Subscriber Ports

There are no internal or external subscriber ports defined for this component assembly.

4.5 External Ports

The Hardware Emulator will service a TCP socket connection with the Adapter component's TCP socket. The socket stream will contain a text message.

5 Component Functionality

At startup, the Adapter component sets an SNA timer (per the SNA Time Management API) to generate a timer event every 5 seconds. Upon timer expiration, the Adapter component's timer callback will check if the socket connection with the Hardware Emulator has been established, it will reset the timer if no connection has been made.

When the Hardware Emulator and Adapter are connected, the Adapter's TCP socket will register the socket file descriptor with the reactor and send a message to the Hardware Emulator to begin sending text messages. The Adapter component's TCP socket is notified from a handler method previously associated with the socket file descriptor to process incoming text messages from the Hardware Emulator.

If a text message is received, it is then passed to the Relay Server component using the provided facet and method `echoText` which takes a CORBA string as a parameter. The text message is then display (per the SNA Logging API).

Once started, this scenario will execute indefinitely, or until the deployment is halted manually.

6 Configurable Parameters

A `HardwareEmulator_bin.cfg` configuration file allows configuration of the host and port to service TCP requests, which is set to "127.0.0.1" and 51214 by default. This file will need to be passed as an argument when launching `HardwareEmulator`. If multiple connections are desired, a configuration file will need to be created with a different port number for each connection.

Since this component assembly is intended to support run-time experimentation, it is also packaged with a full set of SNA compliant run-time execution configuration and deployment files.

7 Design Constraints

1. A Hardware Emulator binary executable will periodically send TCP socket messages.
2. An Adapter Component will connect to the Hardware Emulator and translate information contained in the incoming TCP socket messages to a standard port type.
3. A Relay Server Component will provide a standard port type to allow text messages to be logged.
4. The External Adapter Example component assembly will follow established naming conventions and code organization guidance defined in the SNA SDK documentation, such that new SNA software developers can use it as a design reference.
5. A full set of SNA compliant configuration and deployment files will be provided with this example in order to support run-time execution in a default single-host target deployment.
6. This example will utilize the standard SNA APIs to perform all functions.

8 Component Test

This binary executable with 2-component assembly must be executable on a single “localhost” development computer, to include the SNA SDK x86-64 VM at a minimum. Users can experiment with the deployment plan to redeploy to an alternative 2-host target environment if desired.

9 Component Dependencies

The External Adapter Example assembly is self contained and has no dependencies on any other application components. Its only dependency is on the SNA SDK’s run time execution environment.