

Additional Information

More details about the API as well as a complete set of man pages for over 50 API functions can be found in Altia's API Reference Manual.

Request the manual by calling **(719) 598-4299**, or download it from **www.altia.com**.

Flexibility

Initiating a Connection

The first step in linking an application to an Altia session is establishing a connection to an Altia interface. You can connect to an Altia session that is already running. Alternately, you can choose a function that launches an Altia session, specifies a design file to use, and establishes a connection to that session.

Receiving Events

Altia's API provides three basic methods for receiving notice of an animation variable event (such as a button press or a slider's value change) from the Altia Interface file.

- Polling – The program asks for the current value of a particular animation in Altia.
- Callbacks – Altia provides routines for adding callbacks to specific animations.
- Blocking – Selecting Events and Receiving Events in Queue— Altia allows you to select which events you want to receive. The queue then allows you to process those events in your own time and helps keep them in the proper order.

External Connections

While the API enables communication between the code and the Altia design, the external connections (listed in the External Connections dialog box) gather the information going to and from the Altia design. Programs built with the Altia API can take advantage of these external connections so that specific animations of Altia objects do not have to be accessed.

- External connections make it easy to change what objects your code is connected to without making changes to the code itself.
- While only sending a signal to a single external connection in Altia, you can control multiple objects simultaneously and increase the number of objects you want to control without changing your code.
- All the connections, or signals, going to and from the code are displayed within one dialog box for easy viewing.
- You do not have to know the specific name for each object animation.

For more information contact:

Altia, Inc.
5030 Corporate Plaza Drive
Colorado Springs, CO 80919
Telephone: 719-598-4299
Fax: 719-598-4392
Web: www.altia.com
Email: info@altia.com

© Copyright 2000 by Altia, Inc. All rights reserved.
Altia is a registered trademark of Altia, Inc.
The Altia Eye Logo is a trademark of Altia, Inc.

Altia API

Datasheet

Platforms:
UNIX: Sun, HP, SGI, and IBM
PC: Win 95, 98, 2000, NT

Languages:
C, C++, Visual Basic, Java

Communication
Mechanisms/Protocols:
UNIX: TCP/IP and Domain Sockets
PC: DDE and TCP/IP

Overview

The API (Application Program Interface) connects external code to an Altia interface, offering engineers the best of both worlds. While they can continue to use code to write powerful and complex programs, they now have the option to connect that code to an Altia design. The graphical display created in Altia helps users visualize the code's behavior, and the input events generated from Altia into the code facilitate testing and debugging of the code. The graphical user interface also enables other non-technical team members to test the code. No writing of low-level TCP/IP or socket code is necessary, and no knowledge of communications protocols is required.

The Altia API connects external code to an Altia interface, offering engineers the best of both worlds. The graphical display created in Altia helps users visualize the code's behavior, and the input events generated into the code facilitate testing and debugging.

As an example, an automotive engineer may write code to simulate the exact behavior of a car's HVAC system. He can then develop an Altia interface and use Altia's API to send data between the code and the Altia design. This connection enables him to test and refine the code before a hardware model of the system is even available.

Platforms
 UNIX: Sun, HP, SGI, and IBM
 PC: Win 95, 98, 2000, NT

Languages
 C, C++, Visual Basic, Java

Communication Mechanisms/Protocols
 UNIX: TCP/IP and Domain Sockets
 PC: DDE and TCP/IP

Altia Design and FacePlate can be easily connected with external applications such as Java, simulation models, or compiled programs. Altia's Application Program Interface (API) allows software engineers to establish communications between programs and Altia simulation graphics interfaces.

Simplicity:

This sample pseudocode program, as well as the example written in C below, shows the simple calls that enable a program to listen and wait for an Altia event. When an event occurs, depending upon its value, the program sends new values to different variables in Altia.

```

Connect to Altia session
Choose which events to listen to
While No Communication Errors
{
    Wait for a selected event to occur
    If Event Value is 1
        Send values to animation in Altia
    Else if Event Value is 2
        Send different values to animation in Altia
}
Disconnect from Altia
  
```

The code as it would look written in C:

```

#include <altia.h>
#include <string.h>
void main(int argc, char* argv)
{
    AtConnectId connectId;
    AltiaEventType value;
    int i;
    const char *name;

    connectId = AtOpenConnection(NULL, NULL, argc, argv);

    AtSelectEvent(connectId, "press");

    while(AtNextEvent(connectId, &name, &value) == 0 )
    {
        if (strcmp(name, "press") == 0 && value == 1)
        {
            for(i=0;i<=100;i++)
                AtSendEvent(connectId, "bar", (AltiaEventType) i);
        }
        else if (strcmp(name, "press") == 0 && value == 0)
        {
            for(i=100;i>=0;i--)
                AtSendEvent(connectId, "bar", (AltiaEventType) i);
        }
    }

    AtCloseConnection(connectId);
}
  
```

Architecture Possibilities

