
Coral Data Processor

CDP v 1.0

Documento Preliminar

Julho de 2004

SUMÁRIO

Coral Data Processor v1.0 (CDP 1.0)	4
Métodos	5
Comunicação com o servidor CTI	5
Connect(String IP)	5
Disconnect().....	5
ConnectCoral(Int PortNumber, String Protocol)	5
DisconnectCoral(Int PortNumber)	5
Refresh(Int PortNumber)	5
LogStart(Int PortNumber, String LogFile)	5
LogStop(Int PortNumber)	5
String GetLogFile(Int PortNumber)	5
String GetProtocol(Int PortNumber)	5
Comunicação com o Coral.....	5
Speaker(Int PortNumber)	5
Dial(Int PortNumber, String Numero)	6
Hold(Int PortNumber)	6
Retreive(Int PortNumber)	6
PageQ(Int PortNumber, Long NumPageQ)	6
AnswerPort(Int PortNumber, Long CallIndex)	6
DivertCall(Int PortNumber, String NumDestino, Long CallIndex)	6
ParkPickup(Int PortNumber, String Ramal)	6
ParkCall(Int PortNumber)	6
PressKey(Int PortNumber, Long NumTecla)	6
OffHook(Int PortNumber)	6
OnHook(Int PortNumber)	6
Eventos	7
OnTrunkBusyIdle(Int PortNumber, String NumDevice, Boolean Status)	7
OnStationBusyIdle(Int PortNumber, String NumDevice, Boolean Status)	7
OnCOC(Int PortNumber, Long COC1, Long COC2)	7
OnIncomingCallToACD(Int PortNumber, String CodAgente, String ANI, Long Tronco, Long Grupo)	7
OnIncomingCall(Int PortNumber, Long CallIndex, String ANI, Long Tronco)	7
OnHeldCall(Int PortNumber, Long CallIndex, String ANI, Long Tronco)	7
OnStopCall(Int PortNumber, Long CallIndex)	7
OnTerminateCall(Int PortNumber, String Ramal)	7
OnTimer(Int PortNumber, Long Hora, Long Minutos, Long Segundos)	8
OnConnection(Int PortNumber, Long RamalChamador, Long RamalChamado, Long Status)	8
OnUpdateDisplay(Int PortNumber, String TextoDisplay)	8
OnGeneralMessage(Int PortNumber, String MsgErro)	8
OnAgentStatus(Int PortNumber, Long Grupo, Long Ramal, Long CodAgente, Long StatusAgente)	8

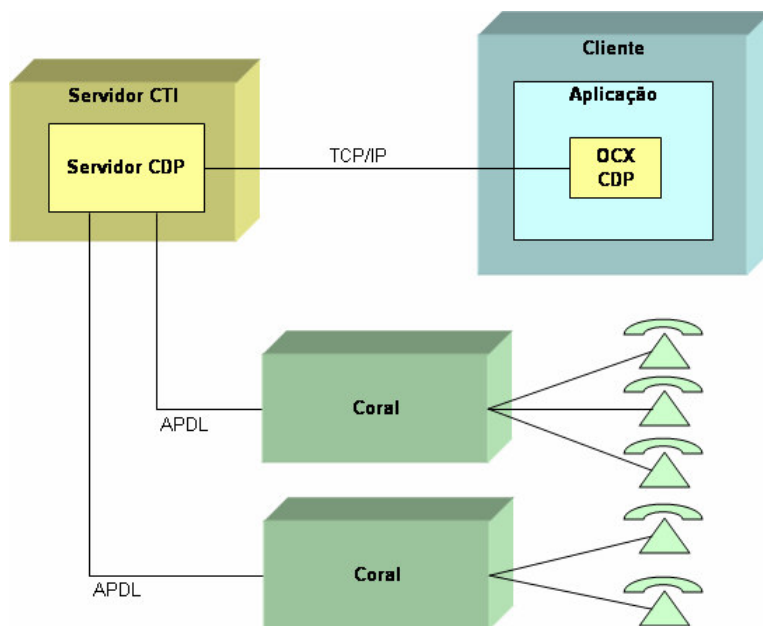
OnWrapUpCode(Int PortNumber, Long Ramal, Long CodAgente, Long CodConclusao)..... 8
OnError(String ErrorMessage, String Proc, Int PortNumber) 8

Coral Data Processor v1.0 (CDP 1.0)

O CDP é um *middle-ware* que permite integrar aplicações ao PABX Coral, através de uma rede TCP/IP, usando a tecnologia COM da Microsoft. A solução é composta de um servidor e um cliente, que podem ser instalados no mesmo computador ou em computadores diferentes, dependendo da capacidade de processamento do computador e do tráfego telefônico do Coral (quanto maior o tráfego maior será o volume de mensagens processados pelo CDP).

Na figura abaixo o servidor CTI é a máquina onde está instalado o Servidor CDP, que se comunica com o Coral através de uma interface APDL, usando uma das portas seriais (COM1, 2, ...). O servidor pode conectar-se a vários sistemas Corais, usando uma porta serial para cada um deles.

O cliente do CDP é o computador onde está instalada a aplicação cliente, por exemplo uma URA ou um serviço de monitoração de *call center*, que captura os status dos agentes e os dados das chamadas em progresso. A aplicação cliente se comunica com o Coral através da OCX CDP, que permite receber mensagens e enviar comandos para o Coral, passando através do servidor CTI. Sempre que a OCX recebe uma mensagem ela deflagra um evento para a aplicação, que pode executar o processamento pertinente. Por exemplo, uma URA pode receber um evento informando que uma chamada foi encaminhada para uma das portas da URA, e capturar os dados pertinentes à chamada (ANI, DNI, tronco, etc.).



Nesta versão o CDP contém apenas recursos *first-party-call-control*, ou seja, ele permite controlar apenas a porta do Coral conectada ao Servidor CTI. Contudo, através deste ramal, ele permite receber mensagens sobre todos os agentes, ramais e troncos conectados ao sistema. A interface entre o Coral e o Servidor CTI é implementada através de um APDL ou de um DKT com interface APA. O APDL geralmente é utilizado quando a aplicação cliente vai apenas receber mensagens do Coral, se a aplicação for atender chamadas, transferir, reter e fazer outros controles, então será necessário utilizar um aparelho DKT.

Métodos

Os métodos são divididos em duas categorias, que são comandos de comunicação com o Servidor CTI e comandos de comunicação com o Sistema Coral.

Comunicação com o servidor CTI

Os métodos de comunicação com o servidor CTI servem para estabelecer e controlar a comunicação com o servidor do CDP.

Connect(String IP)

Faz a OCX conectar-se ao servidor CTI, cujo IP foi informado.

Retorna se a conexão ocorreu com sucesso ou se houveram erros.

Disconnect()

Faz a OCX desconectar-se do servidor.

ConnectCoral(Int PortNumber, String Protocol)

Faz o servidor CTI conectar-se ao Coral, numa porta específica e trabalhando com o protocolo informado (CSTS ou PCC).

DisconnectCoral(Int PortNumber)

Faz o servidor CTI desconectar-se do Coral conectado à porta informada.

Refresh(Int PortNumber)

Reinicializa a comunicação entre o servidor CTI e o Coral, caso esteja ocorrendo erros de comunicação.

LogStart(Int PortNumber, String LogFile)

Faz o servidor CTI logar todas as mensagens recebidas do Coral conectado à porta informada.

LogStop(Int PortNumber)

Faz o servidor CTI parar o log de mensagens recebidas do Coral conectado à porta informada.

String GetLogFile(Int PortNumber)

Retorna um string com o nome do arquivo de log associado ao Coral conectado à porta informada. Se esta porta estiver fechada retorna um *string* vazio.

String GetProtocol(Int PortNumber)

Retorna o nome do protocolo (CSTS ou PCC) utilizado para comunicação com o Coral conectado à porta informada. Se esta porta estiver fechada retorna um *string* vazio.

Comunicação com o Coral

Os métodos de comunicação com o sistema Coral servem para estabelecer e controlar chamadas telefônicas.

Speaker(Int PortNumber)

Equivale a pressionar a tecla “Speaker” do aparelho conectado à porta informada.

Dial(Int PortNumber, String Number)

Disca uma série de números no aparelho conectado à porta informada..

Hold(Int PortNumber, Long CallId, String Device)

Retém a chamada em progresso no aparelho conectado à porta informada.

***Retreive(Int PortNumber)**

Atende a chamada retida no aparelho conectado à porta informada.

***PageQ(Int PortNumber, Long NumPageQ)**

Estaciona ou pega uma chamada na “PageQ” do aparelho conectado à porta informada.

***AnswerPort(Int PortNumber, Long CallIndex)**

Atende a chamada que está “ringando” na porta informada.

***DivertCall(Int PortNumber, String NumDestino, Long CallIndex)**

Desvia a chamada que está “ringando” na porta informada para o número de destino informado.

***ParkPickup(Int PortNumber, String Ramal)**

Atende a chamada estacionada no ramal informado.

***ParkCall(Int PortNumber)**

Estaciona a chamada.

PressKey(Int PortNumber, Long NumTecla)

Pressiona uma tecla do aparelho, para obter o número de cada tecla consulte o manual do Coral.

OffHook(Int PortNumber)

Equivale a tirar o monofone do gancho.

OnHook(Int PortNumber)

Equivale a colocar o monofone no gancho.

Eventos

Descrição dos parâmetros dos eventos:

Parâmetros	Descrição
ANI	Identificação do chamador.
CodAgente	Código do agente.
CodConclusao	Código de conclusão informado pelo agente na finalização de uma chamada.
Grupo	Número do grupo DAC.
MsgErro	Mensagem de erro emitida pelo Coral.
NumDevice	Número do ramal ou tronco.
NumPorta	Número da porta do Coral associada à chamada.
Ramal	Número da porta do Coral.
StatusAgente	Status do agente (logado, em pausa, etc.).
StatusDevice	Status do ramal ou tronco.
TextoDisplay	Texto que aparece no display do ramal digital.
Tronco	Número do tronco do Coral.

OnTrunkBusyIdle(Int PortNumber, String NumDevice, Boolean Status)

Indica que um tronco ficou livre (false) ou ocupado (true), em função do status.

OnStationBusyIdle(Int PortNumber, String NumDevice, Boolean Status)

Indica que um ramal ficou livre (false) ou ocupado (true), em função do status.

OnCOC(Int PortNumber, Long COC1, Long COC2)

Indica o código do texto que aparece no display, sendo que o COC1 está relacionado a linha 1 e o COC 2 a linha do display.

OnIncomingCallToACD(Int PortNumber, String CodAgente, String ANI, Long Tronco, Long Grupo)

Chamada oferecida para um grupo DAC ou agente.

OnIncomingCall(Int PortNumber, Long CallIndex, String ANI, Long Tronco)

Chamada tocando no ramal conectado ao servidor CTI.

OnHeldCall(Int PortNumber, Long CallIndex, String ANI, Long Tronco)

Chamada retida no ramal conectado ao servidor CTI.

OnStopCall(Int PortNumber, Long CallIndex)

Indica que uma chamada foi desconectada.

OnTerminateCall(Int PortNumber, String Ramal)

Indica que uma chamada em conferência foi terminada

OnTimer(Int PortNumber, Long Hora, Long Minutos, Long Segundos)

O APA/VDK envia a hora para a aplicação a cada 1 segundo aproximadamente.

OnConnection(Int PortNumber, Long RamalChamador, Long RamalChamado, Long Status)

Indica que uma chamada foi atendida.

OnUpdateDisplay(Int PortNumber, String TextoDisplay)

Reporta alteração no texto que aparece no display.

OnGeneralMessage(Int PortNumber, String MsgErro)

O Coral enviou uma mensagem de erro ao servidor CTI.

OnAgentStatus(Int PortNumber, Long Grupo, Long Ramal, Long CodAgente, Long StatusAgente)

Reporta “Logon” e “Logout” do agente, assim como “Ready”/”NotReady”.

OnWrapUpCode(Int PortNumber, Long Ramal, Long CodAgente, Long CodConclusao)

Reporta códigos de finalização de chamadas.

OnError(String ErrorMessage, String Proc, Int PortNumber)

Reporta um erro dentro da OCX, que pode ser capturado e tratado pela aplicação.