

# Third Party Integration for Aspect Call Centres Made Simple

Mag. Christian Dombacher (BDD)  
Ing. Wolfdieter Dombacher (BDD)  
Nikolaus Lenaugasse 8  
A-2232 Deutsch-Wagram

03.09.2005

## Contents

1	Introduction and Motivation . . . . .	1
2	Solutions Overview . . . . .	1
3	Winset DDE Bridge . . . . .	3
	3.1 DDE Overview . . . . .	3
	3.2 Implementation Issues . . . . .	5
	3.3 Prototype . . . . .	5
4	Conclusion . . . . .	6

## 1. Introduction and Motivation

The aim of this paper is to present solutions for the integration of a third party application with the Aspect call centre. Several alternatives will be described in their advantages and disadvantages thus forming a base for selection of the appropriate solution.

## 2. Solutions Overview

The nature of screen pop integration dictates the use of an event generating call centre device. Usually these events are tightly coupled to telephony events such

as the ring signal for an approaching call. Based on the available components in a typical Aspect call centre architecture, the following approaches for the integration of a third party application exist

1. *Aspect Winset Adapter* - The Aspect Winset Adapter is a small device replacing the common call centre telephone. In addition to a standard 2-wire telephony termination the Winset Adapter also connects to the agent workstation using a common RS232 Interface. The typical setup is shown in figure 2.1. The agent workstation runs a call control application called *Winset*. Third party applications are integrated using Microsoft's *Dynamic Data Exchange (DDE)*. The Winset Adapter scales very well (per single agent) and is easy to install and maintain. No additional servers are needed, the entire system reliability remains in the same magnitude as with a standard telephony driven *automatic call distribution (ACD)* setup.
2. *Aspect Remote Winset Server* - This solution is targeted to remote and home agents. Agents are integrated in the call centre operation via call back. Using a standard telephone and the Winset client application, the agent may perform the same tasks as a local agent. Please note, that the client application is the same as the one used with the Winset adapter. As before, third party application integration is based on Microsoft's DDE. For the implementation an additional server running the Aspect Remote Winset Server application is required. Network issues have to be considered, as control traffic is not limited to a dedicated interface as is for the Winset adapter.
3. *Aspect CTI* - This solution is based on the *Aspect Application Bridge (ABIG) / Computer Telephony Integration (CTI) Server*, which is part of the Aspect Contact Server portfolio. This solution is required to combine with one of the former two approaches, otherwise no proper handling of logon events is possible. As a consequence, this approach requires a higher amount of hardware resources and should be considered only for an operational Aspect Contact Server environment. For this approach the integration of the third party application is based on server-to-server communication, as the Contact Server lends itself to a central architecture. A separate server instance may be dedicated to application handling thus allowing for a common server-to-client szenario (to be exact - a server-to-server-to-client szenario).

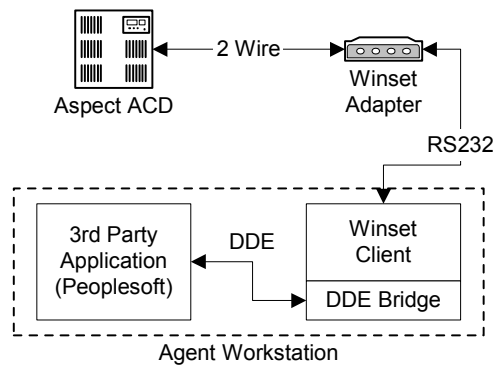


Figure 2.1: Architectural components for the Aspect Winset Adapter configuration

4. Aspect CSS - In case the Aspect *Interactive Voice Response (IVR) System Customer Support Service (CSS)* is in charge of call distribution and agent selection, it may also generate the event which triggers the screen pop. Similar to the CTI solution, a Winset installation is required for proper handling of station control and logon events. This approach should be considered only with a running Aspect CSS and for a server-to-server based communication with the third party application required to pop up.

From the above list, the major differentiators are easily derived to be the coupling method between call centre and third party application (server-to-server or client-to-client), operational devices and business requirements. The following section will focus on the DDE integration, as this coupling method will remain invariant for all client-to-client szenarios.

### 3. Winset DDE Bridge

#### 3.1. DDE Overview

Microsoft *Dynamic Data Exchange (DDE)* is a simple and well understood technology for the exchange of data. It features three types of interaction - *cold*, *warm* and *hot connections*. The difference lies in the nature of data, e.g. cold connections are used for static data only. Warm connections only provide the information, that new data are available. The DDE client is in charge of retrieving

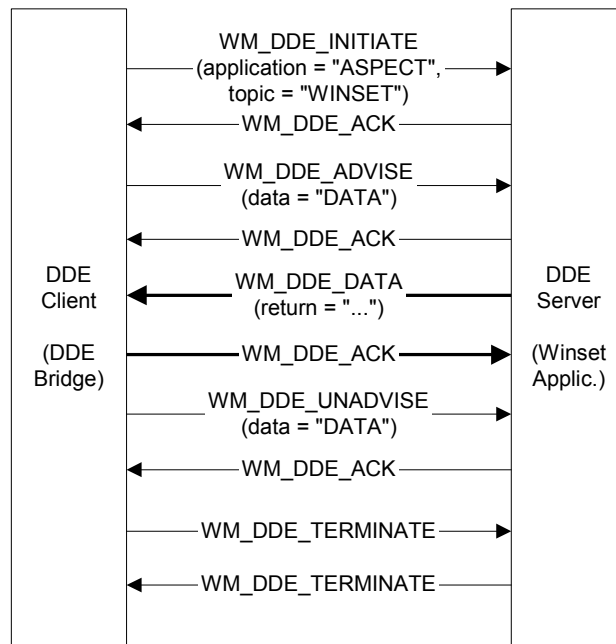


Figure 3.1: Hot connection DDE for Aspect Winset

the most current data by issuing a separate request. With hot connections data are received, as soon as they are available. Aspect recommends a hot connection (see [9]) for the retrieval of so called *cut-through data*, therefore we omit a further discussion on the other connection types. Cut-through data are atomic in the sense, that only a single field may be passed on using DDE. The contents is defined as a property of the *Application* (Display Tab). As the CLI may not be specified directly, Aspect recommends the use of one of the variables A, B, C, D (see [8]). Consequently, the desired variable has to be populated in the *call control table (CCT)* related to the Application. This allows for some flexibility in the choice of contents. Please note, that variable E is not available.

DDE is a message based API (see [1], [2]) and thus follows a simple request and acknowledge pattern. The desired target is addressed via the triple **application**, **topic** and **data**. Aspect has defined the values "ASPECT", "WINSET" and "DATA" for its client application [10]. As already mentioned, "DATA" is atomic and there is no further key to address additional targets. Flexibility has to be carefully planned and incorporated into applications and CCTs. A typical DDE

communication szenario is shown in figure 3.1. The two bold marked messages (**WM\_DDE\_DATA** and **WM\_DDE\_ACK**) are sent with the availability of new data.

For the implementation of DDE driven applications, there exist two approaches - message based DDE and DDEML. The former has been presented above as it forms the base for the latter. DDEML follows the service provider approach and is interfaced using DDEML API functions (see [4]). For more information on DDE, please consult [1], [2], [3], [4], [5].

### 3.2. Implementation Issues

Based on the above technical discussion, a small application called *Winset DDE Bridge* is advocated. It shall implement a DDE client and allow for external program invocation. Configuration settings may be stored in the Windows registry or a separate INI-File. The main task of the DDE Bridge is to wait for new data to arrive and pass them to the external program as a parameter. Depending on the complexity of the application environment, two approaches are possible

1. *Dumb* - Data are passed to the third party application as they are retrieved. In case of a restricted CLI, no data are available. The external application is responsible for data conversion and possible subsequent screen pops to complete the data. This approach is preferable, if only a single third party application exists, as responsibility is not shared between different teams.
2. *Intelligent* - If more third party applications are to be handled by the Winset DDE Bridge, common routines should be implemented as part of it. This reduces the chance for duplicated code and unexpected behaviour. Clearly a higher amount of support is required, as the call centre agent interacts with the DDE Bridge directly.

From an implementation perspective any environment or programming language supporting DDE may be used. There are no compatibility issues with Aspect applications due to the loose coupling exercised by Microsoft's DDE. If minor feedback to the agent is required, a tray icon is recommended.

### 3.3. Prototype

In order to obtain a better imagination of the user perspective of the DDE bridge, we implemented a small prototype. Pictures 3.2 and 3.2 show an implementation as system tray icon. The current calling line identification (CLI) is presented to

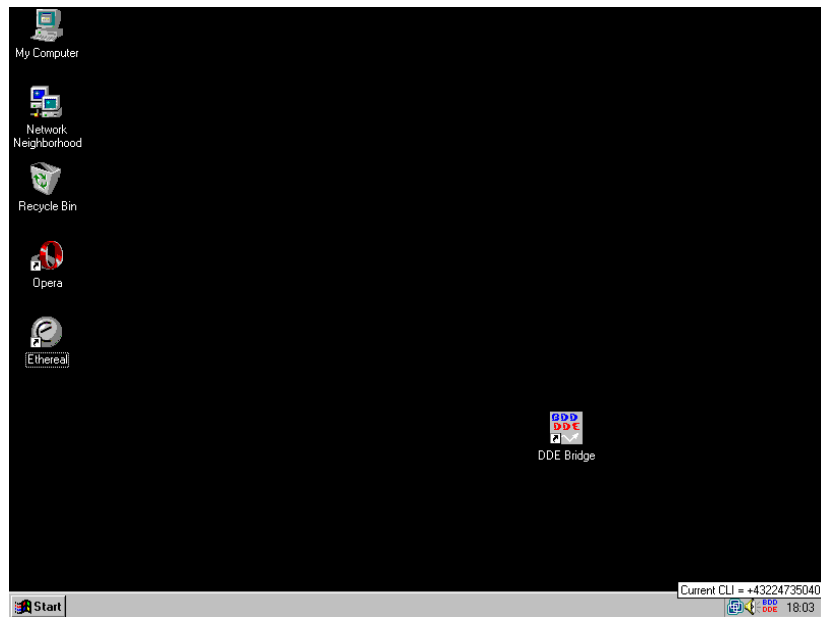


Figure 3.2: Current CLI as a tooltip

the agent as tooltip. The agent only needs to move the mouse pointer over the tray icon. If necessary, application control may be exercised by activating the pop up menu with the right mouse button. The agent may now copy data to the clipboard for use in other non-integrated third party applications. As web driven third party application are commonly used today, the menu also presents an item allowing the agent to launch the default web browser with the third party applications logon page.

#### 4. Conclusion

This paper presented a solutions overview for the integration of third party applications with the Aspect call centre using the technique of screen pop. For smaller configurations, the Aspect Winset Adapter provides a simple and flexible solution. For larger call centre setups, a server based solution should be considered, as these configurations usually architecturally rely on client-server mechanisms and heavy server-to-server coupling. Aspect CTI or an Aspect CSS Integration proves to be feasible only for setups with these components operational. Accordingly a se-

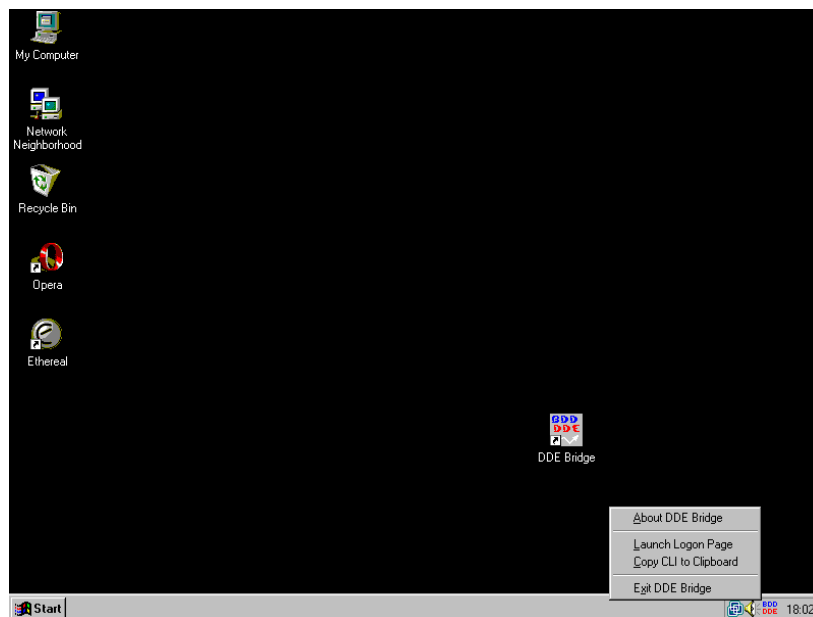


Figure 3.3: Controlling the DDE Bridge

vere amount of development can be expected. For smaller installations this is not necessary and the concept of a DDE bridge provided to be useful. Consequently architectural and developmental information has been included in this paper.

## References

- [1] Richard Wilton - *Microsoft Windows 3 Developer's Workshop* - Microsoft Press 1991
- [2] Charles Petzold - *Programmierung unter Windows 3.1* - Microsoft Press 1991
- [3] WWW - *DDE Overview* - <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winui/winui/windowsuserinterface/dataexchange/dynamicdataexchange/aboutdynamicdataexchange.asp>
- [4] WWW - *DDEML* - <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winui/winui/windowsuserinterface/dataexchange/dynamicdataexchangemanagementlibrary.asp>

- [5] WWW - *DDE FAQ* - <http://www.angelfire.com/biz/rhaminisys/ddeinfo.html>
- [6] Aspect Corporation - *Online Help to Winset 3.1 SDK* - Aspect Corporation 2002
- [7] Aspect Corporation - *Winset 3.1 Client Installation* - Aspect Corporation 2002
- [8] Aspect Corporation - *Aspect Call Center 8.4 User Guide* - Aspect Corporation 2003
- [9] Aspect Corporation - *Winset 3.1 User Guide* - Aspect Corporation 2002
- [10] Aspect Corporation - *Winset 4.0 Planning Guide* - Aspect Corporation 2003