FICS® Real Time Access®
Documentation

RealTimeAccess.htm
Version 2.01

Purpose

FICS® Real Time Access® (RTA) will allow you to make requests for history transactions for loans, requests for loan information, and certain data changes from your own programs. These requests can be made from Tellers Systems, Phone Banking Systems, and Home Banking Systems. The data passed to the function in the Real Time Access® Com Object, RealTimeAccess.DLL, needs to be in an XML style format as described in the included documentation. In order for requests to be handled properly, spelling, capitalization, and syntax must be accurate. The XML that Real Time Access® accepts is patterned after standards that MISMO® (Mortgage Industry Standards Maintenance Organization) uses.

Requirements

There are two main requirements for Real Time Access®:

1. **Real Time Access® must be installed on a Windows® operating system, version 2000 or above.**
2. **An ODBC Data Source must be set up for Real Time Access® to communicate with the FICS® database.**
3. **Microsoft® .NET 2.0 or higher is required for the RTA Socket Service program and its test client. .NET can be installed from the socket service installation program.**

Where does Real Time Access® need to be installed?

Some Important Concepts

- The FICS® Socket Server or Service piece or a self-made web service eliminates the need to install and update Real Time Access® on each client computer in a client-server environment.
- A web application can simply call the Real Time Access® Com Object without using a web service or the FICS® Socket Server or Service.
- A Data Source does not have to be on the same physical LAN as the database server since it can communicate across the Internet with an IP address and an open port on the database server. You should secure the data path so that member sensitive data is encrypted.

Real Time Access® should be installed onto the same PC that the ultimate calling program will be on. Will this program be a non-FICS® program, or the FICS® Socket Server or Service program? This ultimate calling program could be a program at the end of a short list of processes that the request is going through. For example, an Internet client can make a request that goes through a web application which calls a web service, which calls the Real Time Access® Com Object. In that case, the Real Time Access® Com Object needs to be on the PC that is running the web service.

Which PCs or server you install Real Time Access® on depends on which program will be calling it and what type of program it is. There can be many configurations for this. Some are listed here (see pictures below) and others are certainly possible.

Method 1

Method 1 is an example of a teller client-server application calling Real Time Access®. Real Time Access® is installed on each workstation. A drawback of this method is that each workstation has to be set up individually and each workstation will have to be updated when a Real Time Access® release comes out.

Method 2

Method 2 is a variation of Method 1. The teller client-server application is using socket communications to reach the FICS® Socket Server or Service. The Real Time Access® Com Object will be called by the FICS® Socket Server or Service. The socket communication method is probably the most difficult to program. The clients in this example could also call a self-made web service instead of the FICS® Socket Server or Service.

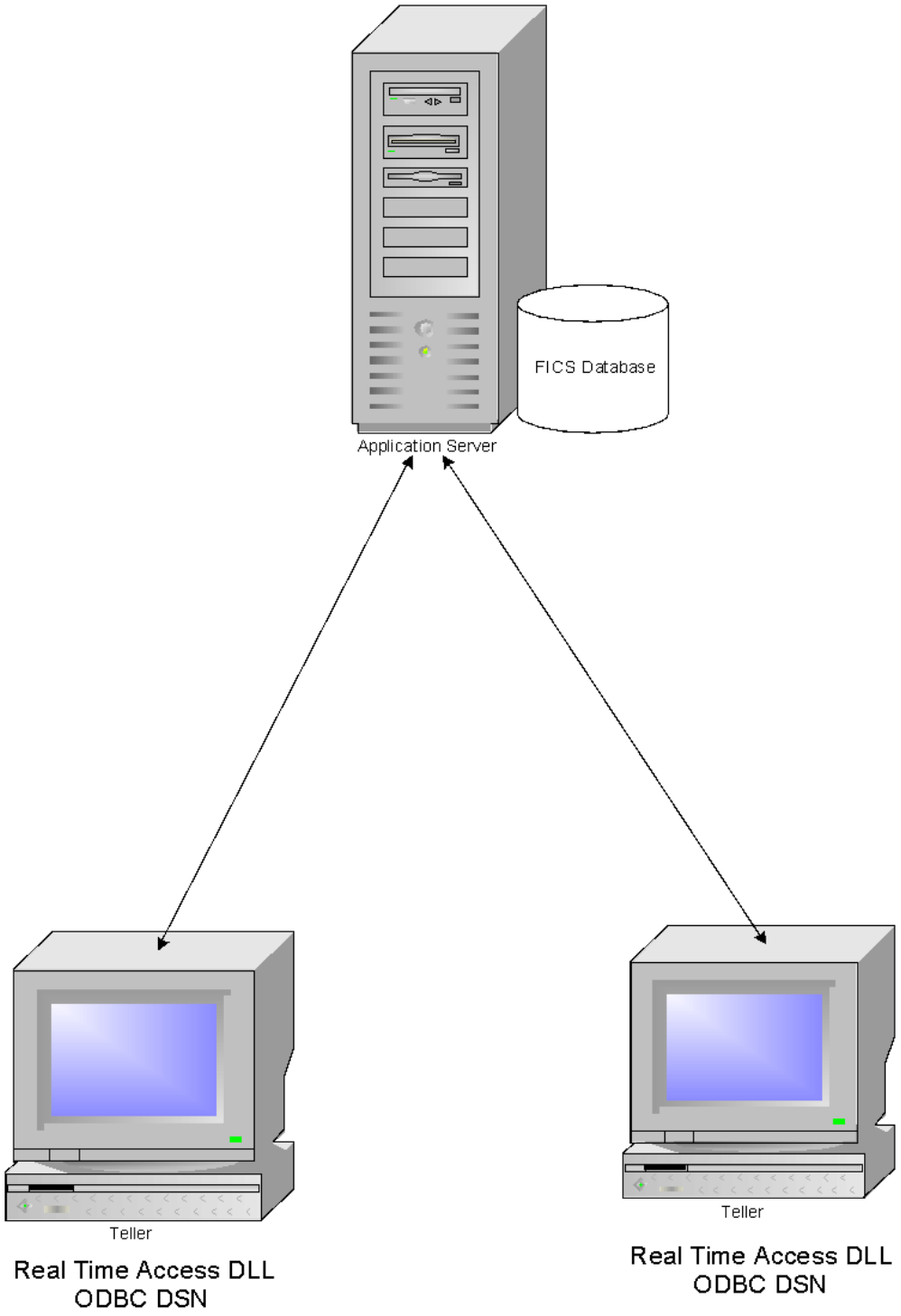
Method 3

Method 3 is a Home/Phone Banking example. If the software that the teller is using was a web application on the same server, the tellers would be using the same installation of Real Time Access® as the borrower.

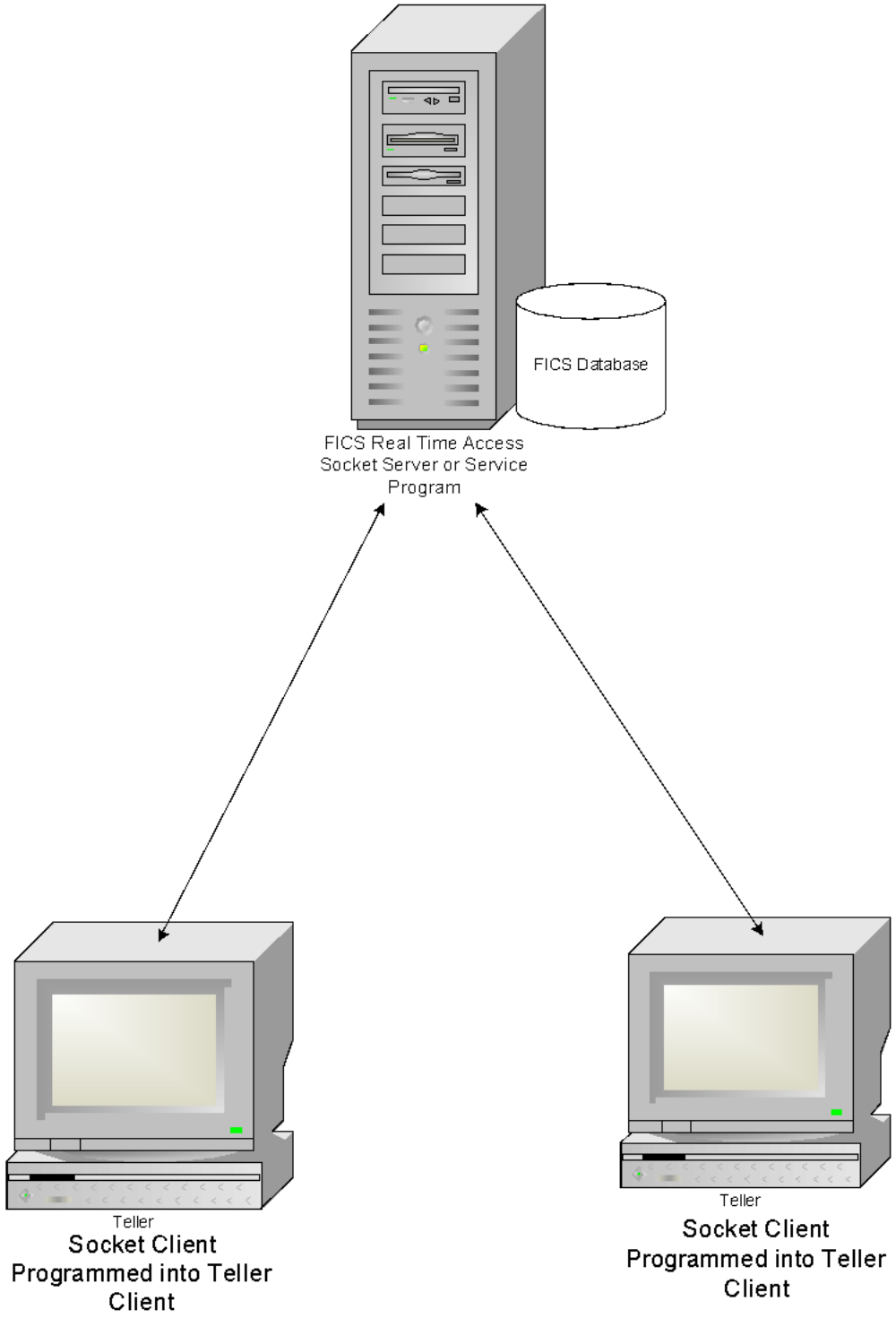
Method 4

Method 4 is a web service example.

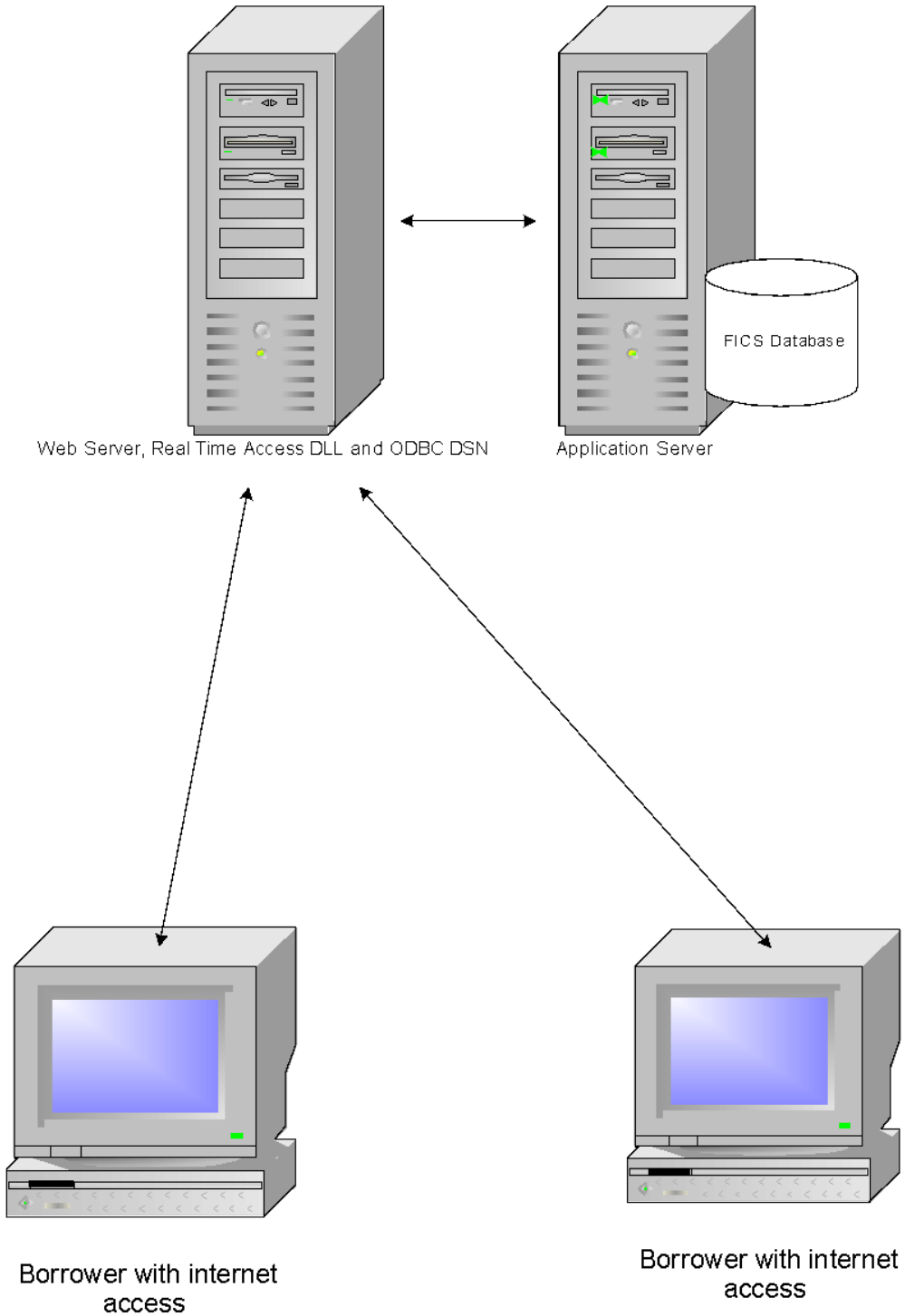
Method 1
(Client-Server Example)



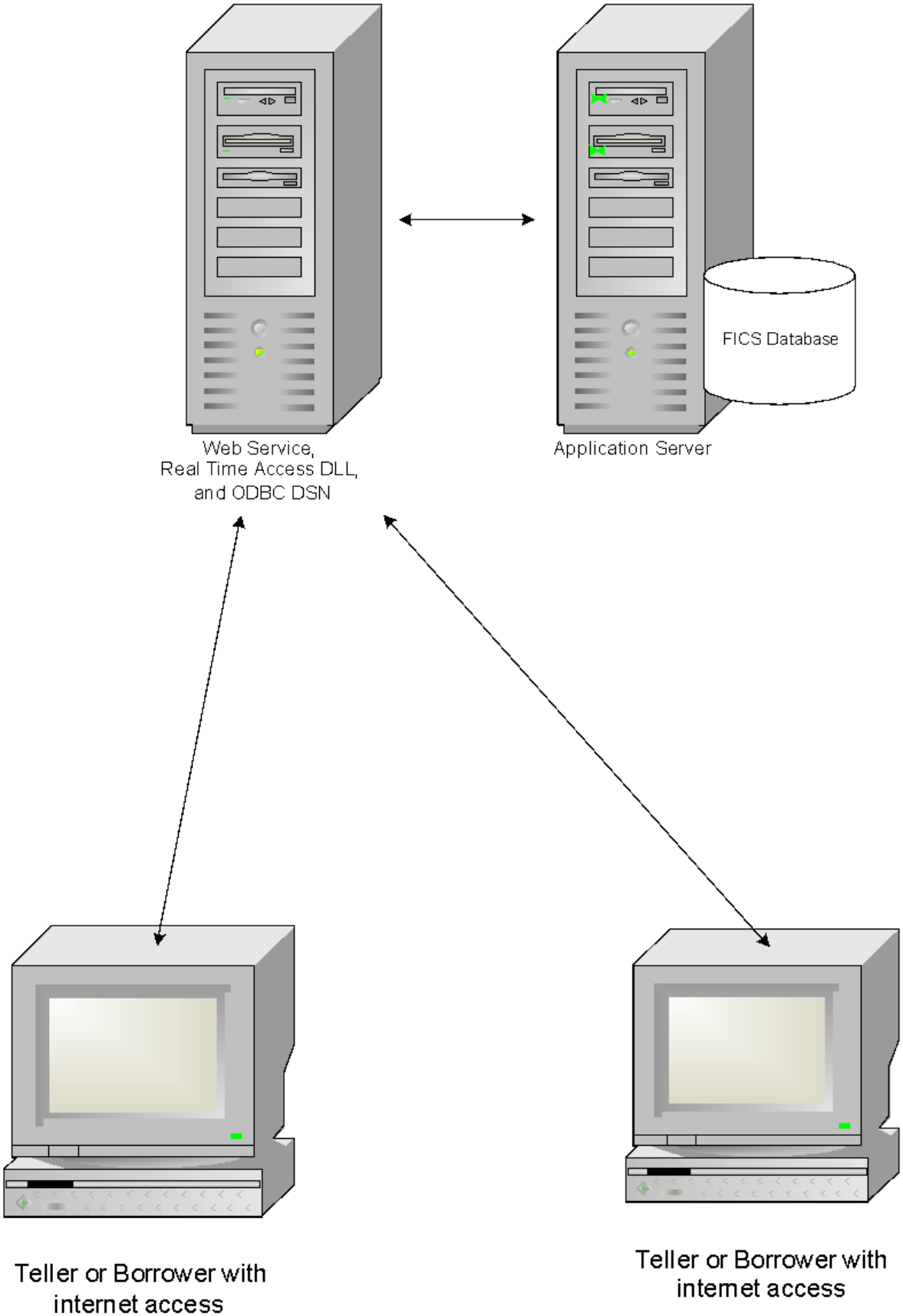
Method 2
(Client-Server Example)



Method 3
(Home / Phone Banking Example)



Method 4
(Web Service Example)



For more information, see the document called Install.doc in the root directory of the CD, and see the documents in the Socket Server and Socket Service folders.

Click this link for a list of files that are on the CD and a description of them.
RealTimeAccess – File List.htm

Requests for Transactions

The history transactions will be made with a "Transaction Description" of "RealTimeAccess" unless the transaction description is supplied in the XML string (see history in the Mortgage Servicer™ system after a payment is made). An exception to this is when a reversal is made. If a transaction description is not supplied, the resulting reversal transaction will have the same description as the transaction that was reversed. After the transaction request is made, updated loan information will be returned to your calling program so that information can be parsed out of the XML string. If a late charge is assessed along with a REGular payment, the "Late Charge Description" in the "Late Charge History" will be "Assess." If a late charge transaction (LTC) is made without a "REG," the "Late Charge Description" in the "Late Charge History" will be "Adjust."

The XML string needed to make a "Transaction Request" needs to be put into a string variable. All of the lines can run together in one long string, but the following example is indented to make it easier to read.

```
Transaction$ = "<?xml version="1.0" ?>
                    <REAL_TIME_ACCESS>
                        <REQUEST_DATA
                    RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                                Requestor="Employee"
                                LoanNumber="1"
                                RequestType="Transaction"
                                OverrideWarning="N">
                            <DEBUG _Mode="Y"
                                _LogFilePath="C:\" />
                            <TRANSACTION _Type="REG"
                                        AmountReceived="1507.5" />
                            <CONNECTION DatabaseType="Sybase"
                                        DataSourceName="DataSourceNameGoesHere"
                                        DatabaseUserID="DatabaseUserIDGoesHere"
                                        DatabasePassword="DatabasePasswordGoesHere"
                                        ApplicationUserID="ApplicationUserIDGoesHere"
                                        ApplicationPassword="ApplicationPasswordGoesHere"/>
                        </REQUEST_DATA>
                    </REAL_TIME_ACCESS>"
```

This is a request to make a regular payment on loan number 1 with the debugging option turned on.

The string variable needs to be passed to the function, f_real_time_access in the DLL.

For example:

```
Result$ = f_real_time_access(Transaction$)
```

An example declaring the Real Time Access® object and making the function call is in each of the .asp files.

The following example shows what was be returned after the above request was made. The information is indented here to make it more readable. Notice how an apostrophe is returned in the DefaultReason attribute of the RESPONSE_DATA element. The rules for handling special characters can be found in the MISMO® document located at this address: http://www.mismo.org/current%20specs/implementationguides.html . After signing in, click the View link for "General Information – Version 2." Once you have the .PDF, look in the XML RESERVED CHARACTERS section of the XML IMPLEMENTATION ISSUES chapter. Even though the "TRANSACTION _Type" was "REG" in the "REQUEST," it was changed to "SDI" for this loan since it is a Daily Interest (Special Daily Interest) loan.

```
<?xml version="1.0" ?>
    <REAL_TIME_ACCESS>
        <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                Requestor="Employee"
                LoanNumber="1"
```

```
                RequestType="Transaction"
                OverrideWarning="N">
        <DEBUG _Mode="Y"
                _LogFilePath="C:\" />
        <TRANSACTION _Type="REG"
                        AmountReceived="1507.5" />
        <CONNECTION DatabaseType="Sybase"
                        DataSourceName="DataSourceNameGoesHere"
                        DatabaseUserID="DatabaseUserIDGoesHere" DatabasePassword="xxxxxx"
                        ApplicationUserID="ApplicationUserIDGoesHere"
                        ApplicationPassword="xxxxxx"/>
</REQUEST_DATA>
<RESPONSE_DATA ResponseDateTime="7/24/03 11:32:19"
                LoanNumber="1"
                Bank="01"
                Investor="006"
                Group="001"
                InterestRate="5.00000"
                PaymentFrequency="Monthly"
                OriginalLoanAmount=".01"
                FundingDate="01/06/03"
                MaturityDate="01/01/33"
                BalloonDate="09/01/03"
                BalloonPayment="500.00"
                NextRateChangeDate="10/01/02"
                NextPaymentChangeDate="07/15/02"
                PayHistory="000000000000"
                DefaultReason="Illness Of Mortgagor&amp;apos;s Family Member"
                DefaultAction="Assumption"
                MemberNumber="member">
        <MESSAGE>
                <_TEXT>Transaction Successful</_TEXT>
        </MESSAGE>
        <PAYMENT_INFORMATION DueDateNextPayment="12/15/03"
                                PrincipalAndInterest="25.00"
                                Subsidy="50.00"
                                TaxAndInsurance="1,312.50"
                                MiscellaneousInsurance="220.00"
                                TotalPayment="1,507.50" />
        <BALANCE Principal="9,935.51"
                Subsidy="67,378.30"
                TaxAndInsurance="1,786.11"
                ForeclosureAndBankFee="11.00"
                UnpaidLateCharges="-796.12"
                MiscellaneousFee="517.54"
                ReturnedCheckCharge="-129.00"
                LossDraftFee="13.00" />
        <HISTORY _ID="1"
                TransactionType="SDI"
                DueDate="11/15/03"
                PaidDate="07/24/03"
                PaymentAmount="1,557.50"
                TransactionDescription="RealTimeAccess"
                UserID=" ApplicationUserID"
                Principal="1,218.47"
                Interest="56.53"
                Subsidy="50.00"
                Unapplied="282.50"
                ServiceFee="90.45"
                ActualDateTime="07/24/03 11:32:19"
                Bank="01"
                Investor="006"
```

```
                    Group="001"
                    InvestorLoanNumber="1234567890"
                    PrincipalBalance="9,935.51"
                    TaxAndInsuranceBalance="1,786.11"
                    SubsidyBalance="67,378.30"
                    LateChargeBalance="-796.12"
                    MiscellaneousFeeBalance="517.54"
                    ForeclosureAndBankFeeBalance="11.00"
                    ReturnedCheckChargeBalance="-129.00"
                    LossDraftFeeBalance="13.00"
                    NegativeAmortizationBalance="65,859.47"
                    PrincipalPaidYearToDate="16,830.29"
                    InterestPaidYearToDate="3,925.29"
                    HazardAndFloodInsurancePaidYearToDate="1,819.00"
                    LateChargesPaidYearToDate="64.00"
                    TaxesPaidYearToDate="9,292.00" />
        </RESPONSE_DATA>
    </REAL_TIME_ACCESS>
```

The following is an example of a Transaction request for a payment reversal. This is a reversal of the payment in the above example. Since the "TRANSACTION _Type" was SDI in the history, an "SDIR" had to be sent as the new type for the reversal to be made.

```
Transaction$ = "<?xml version="1.0" ?>
                <REAL_TIME_ACCESS>
                    <REQUEST_DATA
                RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                            Requestor="Employee"
                            LoanNumber="1"
                            RequestType="Transaction"
                            OverrideWarning="N">
                        <DEBUG _Mode="Y"
                            _LogFilePath="C:\" />
                        <TRANSACTION _Type="SDIR"
                                MatchingDueDate="11/15/03"
                                MatchingDatePaid="07/24/03"
                                MatchingPaymentAmount="1557.50" />
                        <CONNECTION DatabaseType="Sybase"
                                DataSourceName="DataSourceNameGoesHere"
                                DatabaseUserID="DatabaseUserIDGoesHere"
                                DatabasePassword="DatabasePasswordGoesHere"
                                ApplicationUserID="ApplicationUserIDGoesHere"
                                ApplicationPassword="ApplicationPasswordGoesHere"/>
                    </REQUEST_DATA>"
```

The results of calling the function, f_real_time_access, for this reversal are below. (Some data may not look accurate because more payments were made after the one that is being reversed.)

```
<REAL_TIME_ACCESS>
    <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                Requestor="Employee"
                LoanNumber="1"
                RequestType="Transaction"
                OverrideWarning="N">
        <DEBUG _Mode="Y"
                _LogFilePath="C:\" />
        <TRANSACTION _Type="SDIR"
                MatchingDueDate="11/15/03"
                MatchingDatePaid="07/24/03"
                MatchingPaymentAmount="1557.50" />
        <CONNECTION DatabaseType="Sybase"
                DataSourceName="DataSourceNameGoesHere"
```

```
                    DatabaseUserID="DatabaseUserIDGoesHere"
                    DatabasePassword="xxxxxx" ApplicationUserID="ApplicationUserIDGoesHere"
                    ApplicationPassword="xxxxxx"/>
</REQUEST_DATA>
<RESPONSE_DATA ResponseDateTime="7/24/03 15:14:46"
            LoanNumber="1"
            Bank="01"
            Investor="006"
            Group="001"
            InterestRate="5.00000"
            PaymentFrequency="Monthly"
            OriginalLoanAmount=".01"
            FundingDate="01/06/03"
            MaturityDate="01/01/33"
            BalloonDate="09/01/03"
            BalloonPayment="500.00"
            NextRateChangeDate="10/01/02"
            PayHistory="<_span>00000000000"
            DefaultReason="Illness Of Mortgagor&apos;s Family Member"
            DefaultAction="Assumption">
      <MESSAGE>
            <_TEXT>Transaction Successful</_TEXT>
      </MESSAGE>
      <PAYMENT_INFORMATION DueDateNextPayment="11/15/03"
                        PrincipalAndInterest="25.00"
                        Subsidy="50.00"
                        MiscellaneousInsurance="220.00"
                        TotalPayment="195.00" />
      <BALANCE Principal="11,153.98"
            Unapplied="-282.50"
            Subsidy="67,428.30"
            TaxAndInsurance="1,786.11"
            ForeclosureAndBankFee="11.00"
            UnpaidLateCharges="-796.12"
            MiscellaneousFee="517.54"
            ReturnedCheckCharge="-129.00"
            LossDraftFee="13.00" />
      <HISTORY _ID="1"
            TransactionType="SDIR"
            DueDate="11/15/03"
            PaidDate="07/24/03"
            PaymentAmount="-1,557.50"
            TransactionDescription="RealTimeAccess"
            UserID=" ApplicationUserID"
            Principal="-1,218.47"
            Interest="-56.53"
            Subsidy="-50.00"
            Unapplied="-282.50"
            ServiceFee="-90.45"
            ActualDateTime="07/24/03 15:14:46"
            Bank="01"
            Investor="006"
            Group="001"
            InvestorLoanNumber="1234567890"
            PrincipalBalance="11,153.98"
            TaxAndInsuranceBalance="1,786.11"
            SubsidyBalance="67,428.30"
            LateChargeBalance="-796.12"
            UnappliedBalance="-282.50"
            MiscellaneousFeeBalance="517.54"
            ForeclosureAndBankFeeBalance="11.00"
            ReturnedCheckChargeBalance="-129.00"
```

```
                    LossDraftFeeBalance="13.00"
                    NegativeAmortizationBalance="65,859.47"
                    PrincipalPaidYearToDate="15,611.82"
                    InterestPaidYearToDate="3,918.76"
                    HazardAndFloodInsurancePaidYearToDate="1,819.00"
                    LateChargesPaidYearToDate="64.00"
                    TaxesPaidYearToDate="9,292.00" />
        </RESPONSE_DATA>
</REAL_TIME_ACCESS>
```

Requests for General Information

The XML string needed to make a "Transaction Request" needs to be put into a string variable. All of the lines can run together in one long string, but the following example is indented to make it easier to read.

```
GeneralInformation$ = "<?xml version="1.0" ?>
                        <REAL_TIME_ACCESS>
                            <REQUEST_DATA
                        RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                                    Requestor="Employee"
                                    LoanNumber="1"
                                    RequestType="GeneralInformation">
                                <DEBUG _Mode="Y"
                                    _LogFilePath="C:\"/>
                                <CONNECTION DatabaseType="Sybase"
                                        DataSourceName="DataSourceNameGoesHere"
                                        DatabaseUserID="DatabaseUserIDGoesHere"
                                        DatabasePassword="DatabasePasswordGoesHere"
                                        ApplicationUserID="ApplicationUserIDGoesHere"
                                        ApplicationPassword="ApplicationPasswordGoesHere"/
                                        >
                            </REQUEST_DATA>
                        <REAL_TIME_ACCESS>"
```

This is a request for general loan information for loan number 1 with the debugging option turned on.

The string variable needs to be passed to the function, f_real_time_access in the DLL.

For example:

```
Result$ = f_real_time_access(GeneralInformation$)
```

An example declaring the Real Time Access® object and making the function call is in each of the .asp files.

The following example shows what was returned after the above request was made. The information is indented here to make it more readable.

```
<?xml version="1.0" ?>
    <REAL_TIME_ACCESS>
        <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                    Requestor="Employee"
                    LoanNumber="1"
                    RequestType="GeneralInformation">
        <DEBUG _Mode="Y"
                _LogFilePath="C:\"/>
        <CONNECTION DatabaseType="Sybase"
                    DataSourceName="DataSourceNameGoesHere"
                    DatabaseUserID="DatabaseUserIDGoesHere"
                    DatabasePassword="xxxxxx"
                    ApplicationUserID="ApplicationUserIDGoesHere"
                    ApplicationPassword="xxxxxx"/>
        </REQUEST_DATA>
        <RESPONSE_DATA ResponseDateTime = "2/28/03 12:48:55"
```

```
                    LoanNumber = "1"
                    Bank = "01"
                    Investor = "006"
                    Group = "001"
                    InterestRate = "5.00000"
                    PaymentFrequency = "Monthly"
                    OriginalLoanAmount = "100,000.00"
                    FundingDate = "07/01/02"
                    MaturityDate = "07/02/32"
                    BalloonDate = "09/19/03"
                    BalloonPayment = "500.00"
                    NextRateChangeDate = "10/01/02"
                    NextPaymentChangeDate = "07/15/02"
                    PayHistory = "000000000000"
                    DefaultReason = "Illness Of Mortgagor&apos;s Family Member"
                    DefaultAction = "Assumption"
                    MemberNumber = "lksdlk234123"
                    TaxesPaidYearToDate = "9,292.00"
                    HazardAndFloodInsurancePaidYearToDate = "1,819.00">
            <MESSAGE>
                    <_TEXT>Request Successful</_TEXT>
            </MESSAGE>
            <PAYMENT_INFORMATION DueDateNextPayment = "09/15/10"
                                 PrincipalAndInterest = "25.00"
                                 Subsidy = "50.00"
                                 TaxAndInsurance = "1,312.50"
                                 MiscellaneousInsurance = "220.00"
                                 TotalPayment = "1,507.50"/>
            <BALANCE Principal = "-9,629.58"
                    Unapplied = "-5.00"
                    Subsidy = "68,628.30"
                    TaxAndInsurance = "70,541.74"
                    ForeclosureAndBankFee = "11.00"
                    UnpaidLateCharges = "-796.12"
                    MiscellaneousFee = "507.54"
                    ReturnedCheckCharge = "-129.00"
                    LossDraftFee = "13.00"/>
        </RESPONSE_DATA>
    </REAL_TIME_ACCESS>
```

Requests for History

The XML string needed to make a “History Request” needs to be put into a string variable. All of the lines can run together in one long string, but the following example is indented to make it easier to read.

```
History$ = “<?xml version="1.0" ?>
            <REAL_TIME_ACCESS>
                <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                              Requestor="Employee"
                              LoanNumber="1"
                              RequestType="History"
                              HistoryTransactionsRequested="2"
                              StartingDate="02/27/03"
                              EndingDate="02/28/03">
                    <DEBUG _Mode="Y"
                           _LogFilePath="C:\"/>
                    <CONNECTION DatabaseType="Sybase"
                                DataSourceName="DataSourceNameGoesHere"
                                DatabaseUserID="DatabaseUserIDGoesHere"
                                DatabasePassword="DatabasePasswordGoesHere"
                                ApplicationUserID="ApplicationUserIDGoesHere"
                                ApplicationPassword="ApplicationPasswordGoesHere"/>
```

```
            </REQUEST_DATA>
        <REAL_TIME_ACCESS>“
```

This is a request for two history records starting with 02/27/03 for loan number 1 with the debugging option turned on.

The string variable needs to be passed to the function, f_real_time_access in the DLL.

For example:

```
Result$ = f_real_time_access(History$)
```

An example declaring the Real Time Access® object and making the function call is in each of the .asp files.

The following example shows what was returned after the above request was made. The information is indented here to make it more readable.

```
<?xml version="1.0" ?>
    <REAL_TIME_ACCESS>
        <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                Requestor="Employee"
                LoanNumber="1"
                RequestType="History"
                HistoryTransactionsRequested="2"
                StartingDate="02/27/03"
                EndingDate="02/28/03">
            <DEBUG _Mode="Y"
                _LogFilePath="C:\"/>
            <CONNECTION DatabaseType="Sybase"
                    DataSourceName="DataSourceNameGoesHere"
                    DatabaseUserID="DatabaseUserIDGoesHere"
                    DatabasePassword="xxxxxx"
                    ApplicationUserID="ApplicationUserIDGoesHere"
                    ApplicationPassword="xxxxxx"/>
        </REQUEST_DATA>
        <RESPONSE_DATA ResponseDateTime = "2/28/03 14:37:58">
            <HISTORY _ID = "1"
                TransactionType = "REG"
                DueDate = "06/15/10"
                PaidDate = "02/27/03"
                PaymentAmount = "1,557.50"
                TransactionDescription = "RealTimeAccess"
                UserID = "ApplicationUserID"
                Principal = "64.32"
                Interest = "-39.32"
                TaxAndInsurance = "1,312.50"
                MiscellaneousInsurance = "220.00"
                Subsidy = "50.00"
                ServiceFee = "-62.91"
                ActualDateTime = "02/27/03 13:50:24"
                Bank = "01"
                Investor = "006"
                Group = "001"
                InvestorLoanNumber = "1234567890"
                PrincipalBalance = "-9,500.15"
                TaxAndInsuranceBalance = "67,916.74"
                SubsidyBalance = "68,728.30"
                LateChargeBalance = "-796.12"
                UnappliedBalance = "-5.00"
                MiscellaneousFeeBalance = "507.54"
                ForeclosureAndBankFeeBalance = "11.00"
                ReturnedCheckChargeBalance = "-129.00"
                LossDraftFeeBalance = "13.00"
                NegativeAmortizationBalance = "65,859.47"
```

```
                    ReversedFlag = "N"/>
                <HISTORY _ID = "2"
                    TransactionType = "REG"
                    DueDate = "07/15/10"
                    PaidDate = "02/28/03"
                    PaymentAmount = "1,557.50"
                    UserID = "ApplicationUserID"
                    Principal = "64.58"
                    Interest = "-39.58"
                    TaxAndInsurance = "1,312.50"
                    MiscellaneousInsurance = "220.00"
                    Subsidy = "50.00"
                    ServiceFee = "-63.33"
                    ActualDateTime = "02/28/03 10:47:59"
                    Bank = "01"
                    Investor = "006"
                    Group = "001"
                    InvestorLoanNumber = "1234567890"
                    PrincipalBalance = "-9,564.73"
                    TaxAndInsuranceBalance = "69,229.24"
                    SubsidyBalance = "68,678.30"
                    LateChargeBalance = "-796.12"
                    UnappliedBalance = "-5.00"
                    MiscellaneousFeeBalance = "507.54"
                    ForeclosureAndBankFeeBalance = "11.00"
                    ReturnedCheckChargeBalance = "-129.00"
                    LossDraftFeeBalance = "13.00"
                    NegativeAmortizationBalance = "65,859.47"
                    ReversedFlag = "N"/>
                <MESSAGE>
                    <_TEXT>Request Successful</_TEXT>
                </MESSAGE>
            </RESPONSE_DATA>
        </REAL_TIME_ACCESS>
```

Requests for Borrower Information Change

The XML string needed to make a Borrower Information Change request needs to be put into a string variable. All of the lines can run together in one long string, but the following example is indented to make it easier to read.

```
BorrowerInformationChange$ = "<?xml version="1.0" ?>
    <REAL_TIME_ACCESS>
        <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                Requestor="Employee"
                LoanNumber="3"
                SocialSecurityNumber="214140000"
                RequestType="BorrowerInformationChange">
            <DEBUG _Mode="Y" _LogFilePath="C:\" />
            <BORROWER _LastName="Smith" />
            <CONNECTION DatabaseType="Sybase"
                DataSourceName="DataSourceNameGoesHere"
                DatabaseUserID="DatabaseUserIDGoesHere"
                DatabasePassword="DatabasePasswordGoesHere"
                ApplicationUserID="ApplicationUserIDGoesHere"
                ApplicationPassword="ApplicationPasswordGoesHere"/>
        </REQUEST_DATA>
    </REAL_TIME_ACCESS>
```

This is a request to change the last name of the borrower identified by loan number 3 and social security number 214140000 to "Smith" with the debugging option turned on.

The string variable needs to be passed to the function, f_real_time_access in the DLL.

For example:

```
Result$ = f_real_time_access(BorrowerInformationChange$)
```

An example declaring the Real Time Access® object and making the function call is in each of the .asp files.

The following example shows what was returned after the above request was made. The information is indented here to make it more readable.

```
<?xml version="1.0" ?>
    <REAL_TIME_ACCESS>
        <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                Requestor="Employee"
                LoanNumber="3"
                SocialSecurityNumber="214140000"
                RequestType="BorrowerInformationChange">
            <DEBUG _Mode="Y" _LogFilePath="C:\" />
            <BORROWER _LastName="Smith" />
            <CONNECTION DatabaseType="Sybase"
                DataSourceName="DataSourceNameGoesHere"
                DatabaseUserID="DatabaseUserIDGoesHere"
                DatabasePassword="xxxxxx"
                ApplicationUserID="ApplicationUserIDGoesHere"
                ApplicationPassword="xxxxxx"/>
        </REQUEST_DATA>
        <RESPONSE_DATA RealTimeAccessVersion="1.40"
                ResponseDateTime="3/31/04 14:24:56"
                LoanNumber="3">
            <MESSAGE>
                <_TEXT>Request Successful</_TEXT>
            </MESSAGE>
            <BORROWER _FirstName="Fred"
                _LastName="Smith"
                _SocialSecurityNumber="214-14-0000"
                _HomePhoneNumber="( )908-8265" />
        </RESPONSE_DATA>
    </REAL_TIME_ACCESS>
```

Requests for Address Information Change

The XML string needed to make an Address Information Change request needs to be put into a string variable. All of the lines can run together in one long string, but the following example is indented to make it easier to read.

```
AddressInformationChange$ = "<?xml version="1.0" ?>
    <REAL_TIME_ACCESS>
        <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                Requestor="Employee"
                LoanNumber="3"
                RequestType="AddressInformationChange">
            <DEBUG _Mode="Y" _LogFilePath="C:\" />
            <MAILING_ADDRESS _Zip="75244" />
            <CONNECTION DatabaseType="Sybase"
                DataSourceName="DataSourceNameGoesHere"
                DatabaseUserID="DatabaseUserIDGoesHere"
                DatabasePassword="DatabasePasswordGoesHere"
                ApplicationUserID="ApplicationUserIDGoesHere"
                ApplicationPassword="ApplicationPasswordGoesHere"/>
        </REQUEST_DATA>
    </REAL_TIME_ACCESS>
```

This is a request to change the zip code of the borrower's address identified by loan number 3 to "75244" with the debugging option turned on.

The string variable needs to be passed to the function, f_real_time_access in the DLL.

For example:

```
Result$ = f_real_time_access(AddressInformationChange$)
```

An example declaring the Real Time Access® object and making the function call is in each of the .asp files.

The following example shows what was returned after the above request was made. The information is indented here to make it more readable.

```
<?xml version="1.0" ?>
    <REAL_TIME_ACCESS>
        <REQUEST_DATA RequestingCompany="(AssignedByFICS)RequestingCompanyGoesHere"
                      Requestor="Employee"
                      LoanNumber="3"
                      RequestType="AddressInformationChange">
            <DEBUG _Mode="Y" _LogFilePath="C:\" />
            <MAILING_ADDRESS _Zip="75244" />
            <CONNECTION DatabaseType="Sybase"
                      DataSourceName="DataSourceNameGoesHere"
                      DatabaseUserID="DatabaseUserIDGoesHere"
                      DatabasePassword="xxxxxx"
                      ApplicationUserID="ApplicationUserIDGoesHere"
                      ApplicationPassword="xxxxxx"/>
        </REQUEST_DATA>
        <RESPONSE_DATA RealTimeAccessVersion="1.40"
                      ResponseDateTime="3/31/04 15:15:22"
                      LoanNumber="3">
            <MESSAGE>
                <_TEXT>Request Successful</_TEXT>
            </MESSAGE>
            <MAILING_ADDRESS MailingName="Jamie Brown"
                          Address1="123 Floyd Road"
                          _City="Addison"
                          _State="TX"
                          _Zip="75244" />
        </RESPONSE_DATA>
    </REAL_TIME_ACCESS>
```