February 13, 2018

Mitel ACT! Integration Advanced Configuration Guide

Description: This guide provides advanced configuration information so that users and administrators can customize the appearance of the application.

Contents

Introduction

Mitel's ACT! screen pop integration application (Mitel ACT! Up) provides a screen pop and dialing integration to Sage ACT!, a customer/contact management program.

This guide provides advanced configuration information so users and administrators can customize the appearance of the application. It also contains information on debug tracing.

Please see the Mitel ACT! Integration User and Install Guide for normal usage and installation instructions.

(ACT! is a trademark of the Sage North America company. They are not affiliated with Mitel.)

Mitel ACT! Up Configuration

Configuration File

Advanced configuration parameters for Mitel ACT! Up are read from a file named STActUpSettings.xml, which is stored in the application's installation folder (the default installation folder would be C:\Program Files\Mitel\ACT! Integration).

This file is set with reasonable defaults but can be changed if customization is desired. The format of this text file is XML. It is divided into sections related to various application areas.

Changing this file affects all users on that particular client computer (if there are multiple users that regularly log onto the particular machine).

ACT! Data Entities

Various parameters in the settings file reference data items within the ACT! data framework. Given this, it is recommended that the settings file be changed only by someone familiar with ACT! administration and configuration.

The entity names used in the settings file are the ones known as ACT! "Alias" names. ACT! provides a way to list all these items using the ActDiag.exe program. Please refer to your ACT! documentation for details.

There are "special" entity names we've added to easily reference a combination of multiple ACT! entities into one. They begin with "$":

- $BUSINESSADDRESS – Represents all ACT! address entities concatenated to represent a contact's business address.
- $HOMEADDRESS – Represents all ACT! address entities concatenated to represent a contact's home address.

Phone Number Matching

The SearchContactPhoneFields section controls which phone number fields within ACT!'s contact list are to be tested for a match with the connected calling/called number. The default settings for this section, as shown below, will look for a match in the PHONE, MOBILE_PHONE, and HOME_PHONE fields. (Note the ACT! alias name for the contact's business phone is "PHONE".)

```
<SearchContactPhoneFields>

  <SearchContactPhoneField Entity="PHONE"/>

  <SearchContactPhoneField Entity="MOBILE_PHONE"/>

  <SearchContactPhoneField Entity="HOME_PHONE"/>

</SearchContactPhoneFields>
```

Multiple Contact Match Display

When a calling/called phone number matches more than one ACT! contact record, a "Multiple Contact" display will pop up with one row of data per contact. The list of data columns presented in that display are configured with the CustomerMultiMatch section.

Example showing the default settings and resulting example screen:

```
<CustomerMultiMatch>

  <CustomerMultiMatchColumn Label="Contact" Entity="CONTACT" Width="120"/>

  <CustomerMultiMatchColumn Label="Title" Entity="TITLE" Width="150"/>

  <CustomerMultiMatchColumn Label="User?" Entity="IS_USER" Width="50"/>

  <CustomerMultiMatchColumn Label="Company" Entity="COMPANY" Width="225"/>

</CustomerMultiMatch>
```

There is a CustomerMultiMatchColumn line describing each column of ACT! contact data to be displayed. There are 3 parameters for each line in this section. The values of each parameter must be quoted as in the example.

- Label – The value is the string displayed at the top of the column.
- Entity – The ACT! contact data entity alias name whose value should be displayed.
- Width – The width of the column, in pixels.

Contact Detail Display

When a calling/called phone number matches a single ACT! contact record (or the "Contact" name on

“Multiple Contact” display is clicked) the “Contact Detail” display will pop up with a list of rows, each representing a label and entity value from that contact’s ACT! record.

The CustomerDetail section configures the width of each of the 2 columns and the data items (CustomerDetailRow entries) that will make up each row of the display.

Example showing the default settings and resulting example screen:

```
<CustomerDetail LabelWidth="100" EntityWidth="350">

  <CustomerDetailRow Label="Image" Entity="Photo"/>

  <CustomerDetailRow Label="Salutation" Entity="SALUTATION"/>

  <CustomerDetailRow Label="Contact" Entity="CONTACT"/>

  <CustomerDetailRow Label="Title" Entity="TITLE"/>

  <CustomerDetailRow Label="Company" Entity="COMPANY"/>

  <CustomerDetailRow Label="Address" Entity="$BUSINESSADDRESS"/>

  <CustomerDetailRow Label="WebSite" Entity="WEB_SITE"/>

  <CustomerDetailRow Label="Status" Entity="ID/STATUS"/>

  <CustomerDetailRow Label="Birthday" Entity="BIRTH_DATE"/>

  <CustomerDetailRow Label="Create Date" Entity="CREATE_DATE"/>

  <CustomerDetailRow Label="User?" Entity="IS_USER"/>

</CustomerDetail>
```

There are 2 parameters in the CustomerDetail line (the first line of the section). The values of each parameter must be quoted as in the example.

- LabelWidth – The width of the first column, in pixels.
- EntityWidth – The width of the second column, in pixels.

Each CustomerDetailRow has 3 parameters. The first 2 are required, "Highlight" is optional. The values of each parameter must be quoted as in the example.

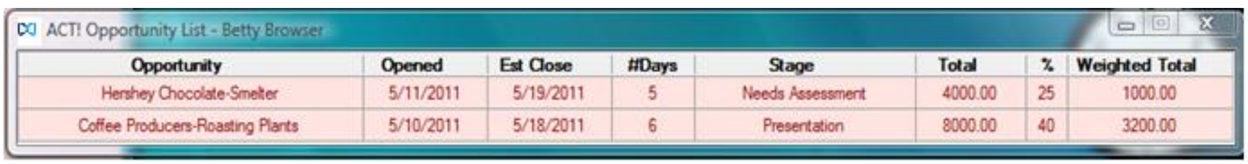
- Label – The value is the string displayed in the first column.
- Entity – The ACT! contact data entity alias name whose value should be displayed in the second column.
- Highlight – Whether this row will be highlighted (in yellow), value is "True" or "False". This parameter may be omitted (defaulting to false, no highlighting). Example:
  <CustomerDetailRow Label="Status" Entity="ID/STATUS" Highlight="True"/>

Opportunity Detail Display

When a "Current Opportunities" link is clicked at the bottom of the contact detail display, an overview display of the ACT! opportunities is shown.

Example showing the default settings and resulting example screen:

```
<OpportunityDetail>

  <OpportunityDetailColumn Label="Opportunity" Entity="OPPORTUNITY_NAME" Width="250"/>

  <OpportunityDetailColumn Label="Opened" Entity="OPEN_DATE" Width="90"/>

  <OpportunityDetailColumn Label="Est Close" Entity="ESTIMATED_CLOSE_DATE" Width="90"/>

  <OpportunityDetailColumn Label="#Days" Entity="DAYS_OPEN" Width="60"/>

  <OpportunityDetailColumn Label="Stage" Entity="STAGE" Width="160"/>

  <OpportunityDetailColumn Label="Total" Entity="TOTAL" Width="80"/>

  <OpportunityDetailColumn Label="%" Entity="PROBABILITY_OF_CLOSE" Width="30"/>

  <OpportunityDetailColumn Label="Weighted Total" Entity="WEIGHTED_TOTAL" Width="120"/>

</OpportunityDetail>
```



For each column there is an OpportunityDetailColumn line describing each column of ACT! opportunity data to be displayed. There are 3 parameters for each line in this section. The values of each parameter must be quoted as in the example.

- Label – The value is the string displayed at the top of the column.
- Entity – The ACT! opportunity data entity alias name whose value should be displayed.
- Width – The width of the column, in pixels.

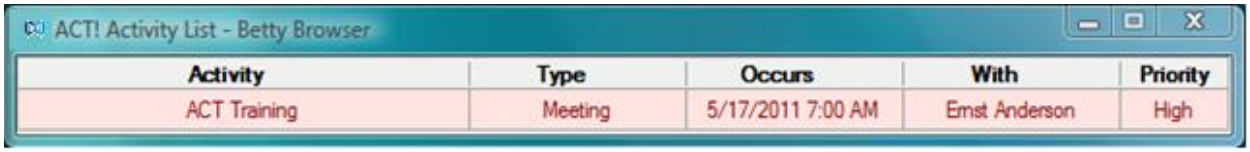
Activity Detail Display

When a “Current Activities” link is clicked at the bottom of the contact detail display, an overview display of the ACT! activities is shown.

Example showing the default settings and resulting example screen:

```
<ActivityDetail>

  <ActivityDetailColumn Label="Activity" Entity="Regarding" Width="250"/>

  <ActivityDetailColumn Label="Type" Entity="TypeName" Width="120"/>

  <ActivityDetailColumn Label="Occurs" Entity="OccurTime" Width="120"/>
```

```
    <ActivityDetailColumn Label="With" Entity="ScheduledWith" Width="120"/>

    <ActivityDetailColumn Label="Priority" Entity="Priority" Width="60"/>

  </ActivityDetail>
```



For each column there is an ActivityDetailColumn line describing each column of ACT! activity data to be displayed. There are 3 parameters for each line in this section. The values of each parameter must be quoted as in the example.

- Label – The value is the string displayed at the top of the column.
- Entity – The ACT! activity data entity alias name whose value should be displayed.
- Width – The width of the column, in pixels.

History Detail Display

When a “Recent History” link is clicked at the bottom of the contact detail display, an overview display of the ACT! contact history is shown.

Example showing the default settings and resulting example screen:

```
  <HistoryDetail>

    <HistoryDetailColumn Label="When" Entity="StartTime" Width="120"/>

    <HistoryDetailColumn Label="History" Entity="Regarding" Width="350"/>

  </HistoryDetail>
```

For each column there is a HistoryDetailColumn line describing each column of ACT! history data to be displayed. There are 3 parameters for each line in this section. The values of each parameter must be quoted as in the example.

Label – The value is the string displayed at the top of the column.

Entity – The ACT! history data entity alias name whose value should be displayed.

Width – The width of the column, in pixels.

Color Coding

All the pop up screens are color-coded based on the ACT! Status values assigned to the particular contact. Each contact within ACT! can be given a status value (this is an ACT! notion…see relevant ACT! documentation for more details).

Each status value can be assigned a foreground color (the color of the text) and a background color (color of the screen panel) in the CustomerColorCodes section of the settings file. Below is the default:

```
<CustomerColorCodes>
  <CustomerColorCode Status="Prospect" ForeColor="DarkBlue" BackColor="Azure"/>
  <CustomerColorCode Status="Customer" ForeColor="DarkGreen" BackColor="Honeydew"/>
  <CustomerColorCode Status="Employee" ForeColor="DarkRed" BackColor="MistyRose"/>
  <CustomerColorCode Status="Family" ForeColor="Sienna" BackColor="LightGoldenrodYellow"/>
  <CustomerColorCode Status="Friend" ForeColor="Sienna" BackColor="LightGoldenrodYellow"/>
</CustomerColorCodes>
```

For each status (for which a color code is desired) there is a CustomerColorCode line describing the color desired. There are 3 parameters for each line in this section. The values of each parameter must be quoted as in the example.

- Status – The name of the ACT! status these colors apply to.
- ForeColor – The text color to be used for screens relating to contacts with this status.
- BackColor – The panel color to be used for screens relating to contacts with this status.

The colors with names are shown at this Microsoft website:
http://msdn.microsoft.com/en-us/library/aa358802.aspx

If a contact has more than one status applied within ACT! (which is allowed by ACT!), the one indicated first is used.

If the contact’s status is not listed in the CustomerColorCodes section of the settings file, then the color will be black text on white panel.

Map Providers

When clicking on a linked address, Mitel ACT! Up tries to open a browser showing a map of that address. It uses one of several web map providers and the MapProviders section of the settings file lists those providers and the format of the URL that must be used to display an address as a map.

Note the choice of which of these providers to use is in the Settings dialog screen as described in the Mitel ACT! Integration User and Install Guide.

As configured in the default settings file, it is shown here for completeness:

```
  <MapProviders>

    <MapProvider Name="Google" URLTemplate="http://maps.google.com/maps?hl=en&amp;source=hp&amp;ie=UTF-8&amp;q=$FULLADDR"/>

    <MapProvider Name="Yahoo" URLTemplate="http://maps.yahoo.com/index.php?ard=1&amp;q1=$FULLADDR"/>

    <MapProvider Name="Mapquest" URLTemplate="http://www.mapquest.com/maps?1c=$CITY&amp;1s=$STATE&amp;1a=$STREETADDR&amp;1y=$COUNTRY&amp;CID=lfddlink"/>

    <MapProvider Name="Bing" URLTemplate="http://www.bing.com/maps/default.aspx?encType=1&amp;where1=$FULLADDR"/>

  </MapProviders>
```

For each provider there is a MapProvider line. There are 2 parameters for each line in this section. The values of each parameter must be quoted as in the example.

- Name – The name of the provider as it will appear in the Mitel ACT! Up settings screen.
- URLTemplate – The template for getting the provider to show a map. This includes subcomponents of the address, positioned where needed, such as $FULLADDR, $CITY, $STATE, $STREETADDR, and $COUNTRY. Each provider requires a different template.

Logging

The application can log errors and potentially debug tracing to a log file.

The application logging is controlled using the file log4net.config. There are 3 different locations that control logging for 3 different components of this application. Each will generate a different log file in the folder: %AppData%\ShoreTel\STActUp (you can type that reference into Windows Explorer to locate the folder).

For each component, logs can be kept up to 7 days (the name of the day is in the log file name).

List of components, location of log4net.config file, and output log file names.

<u>Mitel ACT! Up (main tray application)</u>

Log4net.config location: Our installation folder (C:\Program Files\Mitel\ACT! Integration)

Log file location: %AppData%\ShoreTel\STActUp \STActUp.<day of week>.log

Mitel ACT! Plugin DLL

Log4net.config location: ACT! plugin folder (C:\Program Files\ACT\Act for Windows\Plugins)

Log file location: %AppData%\ShoreTel\STActUp \STActPlugin.<day of week>.log

Mitel ACT! STDial Control

Log4net.config location: ACT! tools folder (C:\Program Files\ACT\Act for Windows\Tools)

Log file location: %AppData%\ShoreTel\STActUp \STControl.<day of week>.log

Note the installation folders shown are the defaults (if not changed by users during installation).

The logging level for each component can be altered by editing log4net.config. There are 2 pertinent levels, INFO and DEBUG. To change, edit the line in log4net.config containing:

```
<level value="INFO" />
```

The value (within quotes) may be INFO or DEBUG. Editing and saving log4net.config causes the logging level to change immediately. It is suggested the level be kept to INFO unless a specific issue is being investigated in which you may be instructed to change the level to DEBUG.