AHLTA Template Documentation (AIM Forms)

Updated: 8/23/2016

Version: Word 2010

# TSWF / Developer Contacts

* POC for form examples: Randy Eccles, MD – TSWF/Customer Deployment Support Branch/DHA; (C) 703-431-6467; [randy.eccles@ehrts.com](mailto:randy.eccles@ehrts.com)
* POC for form designer documentation: Robert Isbell ([Robert.k.isbell.civ@mail.mil](mailto:Robert.k.isbell.civ@mail.mil)) – *no longer working as a TSWF designer*
* TSWF Team (as of March 2014): Dr Brian Jones, CDR James George, LtCol Kevin J Kaps
* TSWF site on milBook (CAC required; <https://login.milsuite.mil/>) - this seems to be the most active discussion forum
* TSWF site on Defense Health Clinical Systems Knowledge Sharing Site (<https://www.ehrknowex.com/groups/tri-service-work-flow>)
* Mark Kline, PA-C – TSWF pediatric lead; cell 240-678-9244
* TSWF Resource Center: <https://www.tswf-mhs.com>

# Exported Structure

## File Structure

|  |  |  |  |
| --- | --- | --- | --- |
| LINE | ITEM | PURPOSE | EXAMPLE |
| 001 | Form Signature | Identifies the designer used to build the form | VB6: “MedcinForm-V1.0”  .NET: “MedcinForm-V1.1” |
| 002 | Form Identification | Consists of four comma-delimited string   * FormName~FormID (optional iteration of template in the designer database) * FormOwner * FormGroup * FormSID (optional security ID) | “MyTemplate~10”,”David Alt”,”System”,”ebb5ba3ab-2799-4bdb-9bda-2108c0dcd3ff” |
| 003 | Form object | Declares the Form object |  |
| 004 | TabStrip object | Declares the TabStrip object, including many template properties |  |
| 005 | BrowseTree object | Declares the BrowseTree object |  |
| 006+ | FormItem (user-determined flags) | Declare the other controls on the form |  |

## Object Structure

Each line represents an object; in most cases, this is a control displayed on the form. These objects are deserialized into comma-delimited strings when a FormTemplate is exported. The order of the exported items doesn’t affect layout (due to absolute positioning) or z-order (seems to be hardcoded).

|  |  |  |  |
| --- | --- | --- | --- |
| POSITION | ITEM | TYPE | DESCRIPTION |
| 00 | Page | int | The page (tab) this item appears on |
| 01 | Left | long | X coordinate of left edge |
| 02 | Top | long | Y coordinate of top edge |
| 03 | Right | long | X coordinate of right edge |
| 04 | Bottom | long | Y coordinate of bottom edge |
| 05 | MedcinID | string | Associated Medcin finding (0 if not associated with a Medcin finding) |
| 06 | Flags | long | Control type |
| 07 | Prefix | string | Control parameters, particularly display/layout |
| 08 | ItemData | string | Additional control parameters, particularly font properties and colors |
| 09 | Description | string | Additional control parameters, particularly caption/text |

Example:

1,10,10,580,30,1718,33562881," |||||||19|440|YCN|0|0|X|X|0|||0|0|0|0|0||||","N=23|T=T","Caption text"

### Flags (long integer)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Flag | Object | Int | Hex | Comments |
| fc\_LEFT | Left alignment, where applicable | 1 | 1 |  |
| fc\_RIGHT | Right alignment, where applicable | 2 | 2 |  |
| fc\_LINE | Line | 1 | 1 |  |
| fc\_BOX | Frame | 2 | 2 |  |
| fc\_TREE | BrowseTree | 4 | 4 |  |
| fc\_GRID | Grid | 8 | 8 |  |
| fc\_RIBBON | Ribbon bar | 16 | 10 |  |
| fc\_TABSTRIP | Page control (tab strip) | 32 | 20 |  |
| fc\_PICTURE | Picture – not implemented in AHLTA | 64 | 40 |  |
| fc\_REQ | *Medcin Finding entry is required* | 128 | 80 |  |
| fc\_CHKY | “Yes” checkbox | 256 | 100 | Medcin object |
| fc\_CHKN | “No” checkbox | 512 | 200 | Medcin object |
| fc\_ROS | ROS state | 1 024 | 400 | Medcin object |
| fc\_VALUE | Textbox to enter value | 2 048 | 800 | Medcin object |
| fc\_BROWSE | Button to browse children findings | 4 096 | 1 000 | Medcin object |
| fc\_NOTE | Button to popup note box for MedcinID > 0. When the flag value is identical to fc\_NOTE and the MedcinID is 0, the record either: (Page=0) the pre-defined chapter headings in the description and itemdata properties (substrings are delimited with a tilde), or (Page=1) the pre-defined Encounter-specific User SubGroup headings in the description and itemdata properties (substrings are tab-delimited) | 8 192 | 2 000 | Medcin object |
| fc\_LINK | Button to link to another form | 16 384 | 4 000 |  |
| fc\_LABEL | Label for captions | 32 768 | 8 000 |  |
| fc\_UPDOWN | UpDown button | 65 536 | 10 000 | Medcin object |
| fc\_HASDATA | A specialized frame with a logic display indicator (LED) in the upper left corner | 131 072 | 20 000 |  |
| fc\_REFERENCE | Reference document, or finding reference button | 262 144 | 40 000 |  |
| fc\_ONSET | Onset entry box | 524 288 | 80 000 | Medcin object |
| fc\_FORM | Form size | 1 048 576 | 100 000 |  |
| fc\_SEARCH | Word search toolbar | 2 097 152 | 200 000 |  |
| fc\_ACTION | Action component | 2 097 152 | 200 000 |  |
| fc\_TEXTBOX | Free text entry box | 4 194 304 | 400 000 | Medcin object |
| fc\_DROPBOX | Picklist dropdown listbox | 8 388 608 | 800 000 | Medcin object |
| fc\_LISTBOX | Picklist full listbox | 16 777 216 | 1 000 000 | Medcin object |
| fc\_LISTVIEW | Picklist multi-column listview | 20 971 520 | 1 400 000 | Medcin object |
| fc\_INLINENOTE | Inline textbox | 33 554 432 | 2 000 000 | Medcin object |
| fc\_RICHTEXTBOX | Rich textbox – not implemented in AHLTA | 37 748 736 | 2 400 000 | Medcin object |
| fc\_PICKLIST | Picklist | 134 217 728 | 8 000 000 |  |
| fc\_ROSTGL | ROS toggle button | 268 435 456 | 10 000 000 | Medcin object |
| fc\_WINDOW | Window – not implemented in AHLTA | 536 870 912 | 20 000 000 |  |
| fc\_TIMING | Timing button component | 1 073 741 824 | 40 000 000 |  |

#### Examples of use

|  |  |  |  |
| --- | --- | --- | --- |
| FLAG (hex) | CONTROL/COMPONENTS | PREFIX[9] | EXAMPLE |
| 1 | fc\_LINE | null |  |
| 2 | fc\_BOX | null |  |
| 4 | fc\_TREE | null |  |
| 8 | fc\_GRID | null |  |
| 16 | fc\_RIBBON | null |  |
| 32 | fc\_TABSTRIP | null |  |
| 256 | fc\_CHKY | YCNH |  |
| 257 | fc\_LEFT, fc\_CHKY | YC |  |
| 513 | fc\_LEFT, fc\_CHKN | YC |  |
|  |  | CY |  |
| 769 | fc\_LEFT, fc\_CHKY, fc\_CHKN | YC |  |
|  |  | CY |  |
| 2305 | fc\_LEFT, fc\_CHKY, fc\_VALUE | YCV |  |
| 4096 | fc\_BROWSE |  |  |
| 4097 | fc\_LEFT, fc\_BROWSE | null |  |
| 4352 | fc\_CHKY, fc\_BROWSE | YCB |  |
| 4865 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_BROWSE | null |  |
| 8193 | fc\_LEFT, fc\_NOTE | YCN |  |
| 8449 | fc\_LEFT, fc\_CHKY, fc\_NOTE | YCN |  |
| 8705 | fc\_LEFT, fc\_CHKN, fc\_NOTE | YCN |  |
| 8960 | fc\_CHKY, fc\_CHKN, fc\_NOTE | YCN |  |
| 8961 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_NOTE | YCN |  |
|  |  | YCBN |  |
|  |  | YCHN |  |
|  |  | YCNH |  |
|  |  | null |  |
| 10497 | fc\_LEFT, fc\_CHKY, fc\_VALUE, fc\_NOTE | YCVN |  |
| 11009 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_VALUE, fc\_NOTE | YCHVN |  |
| 12289 | fc\_LEFT, fc\_BROWSE, fc\_NOTE | CBN |  |
| 12290 | fc\_RIGHT, fc\_BROWSE, fc\_NOTE | CNB |  |
| 13057 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_BROWSE, fc\_NOTE | YCBN |  |
| 16384 | fc\_LINK |  |  |
| 32768 | fc\_LABEL |  |  |
| 32769 | fc\_LEFT, fc\_LABEL |  |  |
| 32770 | fc\_RIGHT, fc\_LABEL |  |  |
| 67841 | fc\_LEFT, fc\_CHKY, fc\_VALUE, fc\_UPDOWN | YCVU |  |
| 68353 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_VALUE, fc\_UPDOWN | YCUV |  |
| 76545 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_VALUE, fc\_NOTE, fc\_UPDOWN | YCUVN |  |
| 131074 | fc\_LEFT, fc\_HASDATA | null |  |
| 131841 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_HASDATA | LYC |  |
| 262144 | fc\_REFERENCE | null | n/a |
| 262401 | fc\_LEFT, fc\_CHKY, fc\_REFERENCE | YCR |  |
|  |  | RCY |  |
| 262402 | fc\_RIGHT, fc\_CHKY, fc\_REFERENCE | YCR |  |
| 533249 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_NOTE, fc\_ONSET | YCON |  |
| 1048576 | fc\_FORM | null | n/a |
| 4202497 | fc\_LEFT, fc\_NOTE, fc\_TEXTBOX | N |  |
|  |  | null |  |
| 4202753 | fc\_LEFT, fc\_CHKY, fc\_NOTE, fc\_TEXTBOX | YCN |  |
|  |  | YCBN |  |
| 4203265 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_NOTE, fc\_TEXTBOX | YCN |  |
| 33562881 | fc\_LEFT, fc\_CHKY, fc\_NOTE, fc\_INLINENOTE | YCN |  |
|  |  | YCBN |  |
| 33563137 | fc\_LEFT, fc\_CHKN, fc\_NOTE, fc\_INLINENOTE | YCN |  |
| 33563393 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_NOTE, fc\_INLINENOTE | YCN |  |
|  |  | NCHY |  |
|  |  | YCNH |  |
| 33825025 | fc\_LEFT, fc\_CHKY, fc\_NOTE, fc\_REFERENCE, fc\_INLINENOTE | RYCN |  |
| 159383553 | fc\_LEFT, fc\_DROPBOX, fc\_LISTBOX, fc\_PICKLIST | null |  |
| 268444417 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_NOTE, fc\_ROSTGL | YCHN |  |
| 268444929 | fc\_LEFT, fc\_CHKY, fc\_ROS, fc\_NOTE, fc\_ROSTGL | YCNH |  |
| 268445441 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_ROS, fc\_NOTE, fc\_ROSTGL | YCHN |  |
| 268449537 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_ROS, fc\_BROWSE, fc\_NOTE, fc\_ROSTGL | YCBNH |  |
| 268512257 | fc\_LEFT, fc\_ROS, fc\_VALUE, fc\_NOTE, fc\_UPDOWN, fc\_ROSTGL | CUVHN |  |
| 268513025 | fc\_LEFT, fc\_CHKY, fc\_CHKN, fc\_ROS, fc\_VALUE, fc\_NOTE, fc\_UPDOWN, fc\_ROSTGL | YCUVHN |  |
| 268968961 | fc\_LEFT, fc\_ROS, fc\_NOTE, fc\_ONSET, fc\_ROSTGL | COHN |  |

### Prefix

The “Prefix” is a pipe-delimited string with many control properties, many of which affect display and layout of the control.

"0 |1|2|3|4|5|6| 7| 8| 9|10|11|12|13|14|15|16|17|18|19|20|21|22|23|24|25"

#### Structure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| POS | TYPE | TERM | PURPOSE | EXAMPLES |
| 0 | string | Prefix | Adds a prefix to the finding in the note output and, depending on the prefix and finding, may override default placement in the note. Additional details and prefix codes listed separately. Maximum of 2 characters.  May also contain a ListSID for PickList objects, if present. | AL: “Allergy to <finding>: custom text”  *(tested prefix codes documented in a separate table)* |
| 1 | string | Modifier | When a non-blank modifier is selected and the finding entered at runtime, it will be entered with that modifier. However, the modifier is not used as a property for matching when populating a form with an existing encounter. | EX, GD, FR, PR for wellness levels; *(large list exists, not yet included in this documentation)* |
| 2 | string | Result | The result qualifier associated with the finding | abnormal, normal, low, high |
| 3 | string | Status | The status qualifier associated with the finding | agreed to, expanding, louder, refused, stable; *(large list exists, not yet included in this documentation)* |
| 4 |  | Value | default value if a value/updown box is used | TSWFPGEN |
| 5 | string | LinkGroup | Link grouping is a mechanism to group together a set of hierarchically related findings for presentation in the Medcin narrative. Entry is limited to a single character (1..9, A..Z).  *Also used by PickList objects (unsure how)* | 1..9, A..Z, a..z |
| 6 | string | Units | Certain controls display this after the label | cm, days |
| 7 | int | BoxOffset | Offset for a multitextbox, or less formally, the height of the caption component (with the multitextbox immediately adjacent below the caption).  Used when the finding has either a list box style for the browse button, or text box style for the note button; its value is adjusted by use of the spinner.  *Also used by PickList objects (unsure how)* | usually 19 |
| 8 | int | InlineTextboxWidth | Width of textbox (using right edge as start point); default is 80 pixels, maximum width is 200 pixels. | usually 80 |
| 9 | string | CustomComponentSequence | Determines visual order of component parts/widgets | YCN |
| 10 | ? | IndexToReferenceList | URL reference number  If your form has a reference list, you can assign a particular reference to the finding by selecting it from the list of available findings displayed when you click the “**Select Reference**” button. Note that only references that you have identified to be used for findings (by specifying a logic string of “#nnn”) will appear in the selection list. The number of the reference that you select followed by the reference description enclosed in parentheses will appear in the box. | 0 for null |
| 11 | ? | NarrativeGroupAssignment | You can pre-assign the narrative group that the finding will be placed into by the narrative engine if you select something other than “*Medcin Default*” group from the “**Narrative Group Assignment**” drop down list. Note that a selection other than “Medcin Default” will override any group selection of a list when that finding is used to invoke a form browse.  Chapter numbers in the forms designer are 0-based. This means that when chapter assignments are not made, all findings are placed in chapter 0. When this occurs, the Medcin narrative engine does not display a chapter header. Chapter numbers can be assigned from 0 to 100. When the narrative engine emits a numbered header for a chapter, the number displayed is one greater than the assigned number so that chapter 0 is displayed as chapter 1. Keep this in mind when assigning your chapter numbers. | 0 seems to be the default value |
| 12 | char | “yes” control caption | The character displayed inside a “Yes” checkbox | X, Y, N, A |
| 13 | char | “no” control caption | The character displayed inside a “No” checkbox | X, Y, N, A |
| 14 | int | BitFlags | * bits 0 – 1: automatic browse * bit 2: disable HPI-ROS toggle * bit 3: disable gender check * bit 4: ValueSum | “8” for EPDS appears to have disabled gender check |
| 15 | int | LimitMax | Maximum value permitted in a value box or UpDown component | “10” for pain-level item (with Description: “:0:10:1:Pain level”) |
| 16 | int | LimitMin | Minimum value permitted in a value box or UpDown component | “0” for pain-level item (with Description: “:0:10:1:Pain level”) |
| 17 | int | RibbonTriggerID | When object is activated, trigger expansion of the specified ribbon.  The “**Ribbon Trigger**” setting determines if this finding acts as a trigger for a ribbon’s collapse and/or expansion. The drop down list will contain the list of ribbons on your form. Choosing something other than “None” will assign the finding as a trigger. Note that more than one finding can be a trigger of the same ribbon. |  |
| 18 | int | ClusterID | A finding can be assigned to a “**Cluster**” of findings for use with the Active Forms module. Typically a single finding acts as a trigger to disable or enable all other findings that are in the same cluster. This allows you to specify a master finding that could keep the remaining cluster findings disabled if it is entered negative. The options allow you to disable the cluster on either an entry of yes or an entry of no. Checking both is permitted. Clusters are form global. This means that a cluster is not limited by page; the same cluster can be spread across multiple pages and the trigger finding can be on any page. Note that a finding can be assigned to a radio button group or a cluster but not to both. Note that you can limit the number of form findings that can be entered within a frame component by using frame logic. | Used to group components, often with O=25 for exclusivity (e.g. radio button selections).  This can be used to have multiple versions of a single Medcin finding without throwing an error – assign the items the same code and the same ClusterID and use O=25; the others will be disabled when any one of them is clicked. |
| 19 | int | ParentRibbonID | Assigns the specified ribbon as this object’s parent ribbon. |  |
| 20 | int | RadioButtonGroup | The finding can be made part of a radio button group by selecting a non-zero “**Radio Button Group**” value. All findings having the same radio button group value on the same page are considered part of the same group. When used in the Active Form’s module, only one of the findings in the group will be permitted entry; if another is entered, any other finding in the group that is entered will be automatically deleted. When the radio button group is first set to a non-zero value, the style of the Yes and No components will be set to the corresponding radio button style if they are currently using a check box style. Note that a finding can be assigned to a radio button group or a cluster but not to both. |  |
| 21 | ? | ImageID | The “Imager Link” tab is used to provide a reference to an image assigned to an imager control object. With this assignment, the corresponding image will be rendered into the imager control when this finding becomes selected. |  |
| 22 | ? | HotSpotSetIdentifier |  | Usually null – have also seen -1 |
| 23 | int | ParentFrame | Identifies the parent frame for all objects other than boxes and grids. | Always null |
| 24 | ? | CodeMapping | You can select a code assignment by pressing the “Select Code Assignment” button. A dialog listing all the codes available for the finding’s Medcinid will appear allowing you to choose one. The selected code will then appear in the text box to the right of the button; the numeric code set type will be prefixed as shown in the illustration above. ICD-10-CM codes may have additional selections for digit positions pertaining to specifier lookup table selections; they are available when the Flags column contains an “S” character. The “Select Specifier” button will be enabled when the selected row indicates availability of a specifier. In the example shown, the button will be enabled when the M79.66 row is selected. The following dialog will be shown upon clicking the button:  Alternatively, you can directly type into the text box to provide your own description that can be used in the narration based on the Narrative Format preference. | Always null |
| 25 | ? | UserAssignedSubgroup | “**Misc SubGroup**” box allows you to assign a user subgroup number to the finding. This property is available only if the Narrative Group Assignment is one of the miscellaneous narrative groups. You can choose any number even if you have not specified a user subgroup heading caption. Note that user subgroup heading captions can be encounter specific, connection specific, or server global – depending on how your application chooses to support miscellaneous user subgroups. Even if the subgroup number you assign here has no corresponding user subgroup header caption, the Medcin server will provide a default caption of “Misc SubGroup n”, where “n” is the subgroup number you assign. If you assign a non-zero user subgroup number and subsequently change the Narrative Group Assignment to a standard Medcin group, the user subgroup number will be automatically reset to 0. | Always null |

#### Prefix Codes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Prefix  (2 char) | Major Group  (if forced) | Subsection | Attached Phrase | Comments |
| (blank) |  |  |  |  |
| ? |  |  | possibly |  |
| ?+ |  |  | probable |  |
| A | Assessment |  | admission diagnosis of |  |
| AD |  |  | administered |  |
| AG |  |  | authorization given for |  |
| AL | Allergies |  | allergy to |  |
| AO | Assessment |  | (none) |  |
| AV |  |  | avoid |  |
| C |  |  | carrier of |  |
| CN |  |  |  |  |
| CO |  |  | order canceled for |  |
| CT |  |  | continue |  |
| D |  |  | cause of death was |  |
| DC |  |  |  |  |
| DI | Discussed |  | discussion of |  |
| DN |  |  |  |  |
| DR | Discussed |  | discussion of risk of |  |
| DS | Assessment |  | discharge diagnosis of |  |
| E |  |  | evaluate preoperatively for |  |
| ED |  |  |  |  |
| EN |  |  |  |  |
| F | Family History |  | (none) |  |
| F0 | Family History | Paternal | (none) |  |
| F1 | Family History | Maternal | (none) |  |
| F2 | Family History | Paternal grandfather’s | (none) |  |
| F3 | Family History | Paternal grandmother’s | (none) |  |
| F4 | Family History | Maternal grandfather’s | (none) |  |
| F5 | Family History | Maternal grandmother’s | (none) |  |
| F6 | Family History | Fraternal | (none) |  |
| F7 | Family History | Sororal | (none) |  |
| F8 | Family History | Son’s | (none) |  |
| F9 | Family History | Daughter’s | (none) |  |
| FB | Family History | Baby’s father’s | (none) |  |
| FC | Family History | Cousin’s | (none) |  |
| FJ | Family History | Paternal aunt’s | (none) |  |
| FK | Family History | Paternal uncle’s | (none) |  |
| FL | Family History | Maternal aunt’s | (none) |  |
| FM | Family History | Maternal uncle’s | (none) |  |
| FP |  |  |  |  |
| FU | Plan |  | follow-up |  |
| H | Past medical/surgical history |  | history of |  |
| HR |  |  | recurrent history of |  |
| HU |  |  | uncorroborated history of |  |
| HY |  |  |  |  |
| IB |  |  |  |  |
| IL |  |  |  |  |
| IN | Allergies |  | intolerant to |  |
| IO |  |  | intra-operative |  |
| IP |  |  | intraprocedural |  |
| IR |  |  |  |  |
| IS | Counseling/Education |  | printed information given for |  |
| IT |  |  |  |  |
| M |  |  |  |  |
| MD |  |  | modify dose of |  |
| MO |  |  |  |  |
| NK |  |  |  |  |
| O |  |  | ordered |  |
| O1 |  |  |  |  |
| O2 |  |  |  |  |
| O3 |  |  |  |  |
| OC |  |  | ordered repeat of |  |
| OP |  |  | order postponed for |  |
| OX |  |  | ordered prophylactic |  |
| P | Assessment |  | primary diagnosis of |  |
| P1 |  |  |  |  |
| PC |  |  |  |  |
| PD |  |  | performed |  |
| PE |  |  | perioperative |  |
| PF |  |  | prophylactic |  |
| PH | Past medical/surgical history |  | prior hospitalization for |  |
| PO |  |  | post-operative |  |
| PP |  |  | postprocedural |  |
| PR |  |  |  |  |
| PS |  |  | post |  |
| PT |  |  |  |  |
| PU |  |  |  |  |
| R |  |  | reviewed |  |
| RC |  |  |  |  |
| RD | Referred here |  | referral diagnosis of |  |
| RE | Past medical/surgical history | Reported: |  |  |
| RF |  |  | referred elsewhere for |  |
| RI |  |  | risk of |  |
| RM |  |  |  |  |
| RO | Rule out |  | rule out |  |
| RS |  |  |  |  |
| RV |  |  | reviewed PMH for |  |
| RX |  |  | renew |  |
| S |  |  | secondary diagnosis of |  |
| SC |  |  | scheduled for |  |
| TF |  |  |  |  |
| UN |  |  | unresponsive to |  |
| VR | Reason for Visit |  | visit for |  |
| W |  |  | working diagnosis of |  |

* Unlisted (valid) prefixes do not cause errors and do not affect term placement – this can be used to differentiate identical MedcinIDs without affecting the note output.
* If the prefix set in the template is not blank, the user will not be able to change it at runtime.
* R# and E# seem to be safe to use without affecting placement

**MAJOR GROUPS, in order of initial default sequence**

Global Const G\_REASON = 1 ' Chief complaint

Global Const G\_REFERDX = 20 ' Referred here

Global Const G\_HPI = 2 ' History of present illness

Global Const G\_CURMED = 3 ' Current medication

Global Const G\_PAST\_MED\_HIST = 4 ' Past medical history

Global Const G\_PAST\_THERAPY = 19 ' Previous therapy

Global Const G\_PER\_HIST = 5 ' Personal history

Global Const G\_HIST = 6 ' History

Global Const G\_FAM\_HIST = 7 ' Family history

Global Const G\_ROS = 16 ' Review of systems

Global Const G\_PHYSICAL = 8 ' Physical findings

Global Const G\_OBSERV = 9 ' Objective

Global Const G\_TESTS = 10 ' Tests

Global Const G\_PAST\_TESTS = 18 ' Previous tests

Global Const G\_ASSESS = 11 ' Assessment

Global Const G\_TESTCON = 21 ' Test conclusions

Global Const G\_THERAPY = 12 ' Therapy

Global Const G\_ALLERGYS = 24 ' Allergies

Global Const G\_VACCINES = 23 ' Vaccinations

Global Const G\_EDUCATION = 22 ' Counseling/Education

Global Const G\_DISCUSS = 25 ' Discussed

Global Const G\_PLAN = 13 ' Plan

Global Const G\_NOTE = 14 ' Notes

Global Const G\_OTHER = 15 ' Other

Global Const G\_EVAL = 17 ' Practice management

Global Const G\_CAREPLAN = 99 ' Care Plan

### ItemData (string)

Contains options, properties, and logic specific to the type of control. Each is separated by a “|”, and logic expressions must always be last (when included). If a control has option flags specific to it, they will be listed separately under that control.

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| BackColor | K=<int> | Specifies the back color of the object (integer value of BBGGRR) |
| BackStyle | T=<0|1> | Specifies whether an object is transparent (0) or opaque (1) |
| BorderStyle | S=<0|1> | Specifies an object’s border style (0=?, 1=?) |
| DrawingDrawWidth | W=<int> | Specifies the thickness of a Drawing’s lines |
| FontBold | B=<T|F> | Specifies whether the object’s caption is **bold** |
| FontItalic | I=<T|F> | Specifies whether the object’s caption is *italicized* |
| FontName | F=<string> | Specifies the font used for the object’s caption; *default is Microsoft San Serif … I think!* |
| FontSize | Z=<8..36> | Specifies the font point size for a label (8 – 36 points) |
| FontUnderline | U=<T|F> | Specifies whether the object’s caption is underlined |
| ForeColor | C=<int> | Specifies the fore color of the object (integer value of BBGGRR) |
| Id | N=<int> | Specifies a unique numeric ID; controls are given an Id sequentially, but the value is not used internally |
| Logic | L=<string> | Specifies the logic equation or other list, if any, and is always last. It may contain a list of MedcinIds for use with an internal list. It may contain a list of ImageIds for use with an Imager object |
| Options | O=<int> | Specifies additional options via a combined flag value (see below) |
| Tooltip | P=<string> | Specifies the tooltip for objects with captions |

### Description (string)

Contains optional packed data and the object’s caption. For most objects, this is a string in the form “Caption~Content-Line1~Line2~Line3”

# Components

## Line (1)

An fc\_Line object has drawing data packed into the description property. This includes fore color, draw style, fill style, and draw width. These are delimited by a colon. The line caption, if any, follows (e.g. “0:0:1:2:This line has a caption without an arrowhead”). Fill color is always the same as fore color. For lines, fill style determines if the line has an arrow head. For horizontal lines, 5 designates a left arrow head and 4 designates a right arrow head. For vertical lines, 5 designates an up arrow head and 4 designates a down arrow head.

To make a line match the same color as a frame, use 12566463 (soft grey).

## Frame (2)

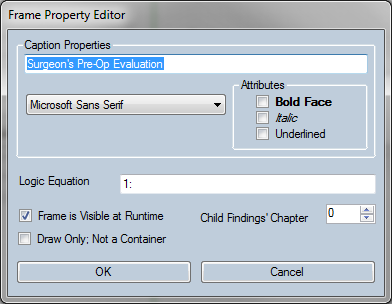
Draw style properties are packed in the Description field in the format <Color>:<OutlineStyle>:<FillStyle>:<DrawWidth>:<Caption>

When the fore color is 0, the box is rendered as a standard frame and may include a caption, a logic equation, and an LED component.

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| FrameChapterAssignment | H=<?> | Specifies the chapter assignment of a Frame |
| FrameDrawOnly | T=<?> | Specifies whether a Frame is draw-only (probably T|F) |

Example:

1,10,10,110,110,0,2,"|||||||0|0||0|0|||0|||0|0|0|0|0|0|||","T=T","0:0:0:1:My Caption"



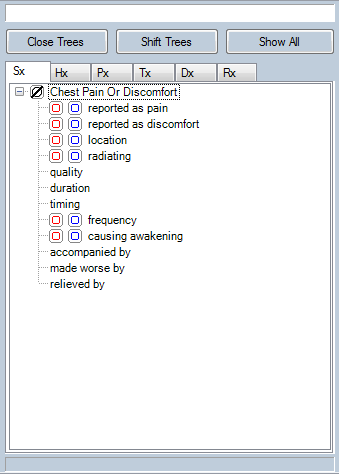
*Frame property editor from the Forms Designer*

## BrowseTree (4)

1. For an fc\_Tree object, the presence of fc\_Left or fc\_Right flag bits specify the global setting of left or right lateral only attributes for tree browsing.

The BrowseTree is common to the form, not for each page.

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| BrowseTreeIntelligentPrompting | I=<T|F> | Specifies the availability of intelligent prompting for a BrowseTree |
| BrowseTreeMouseBrowsing | B=<T|F> | Specifies the availability of mouse browsing for a BrowseTree; if false, the cursor will not change when hovering over a term and the user will not be able to open the BrowseTree by right-clicking |
| BrowseTreeWordSearch | S=<T|F> | Specifies the availability of word search for a BrowseTree |



*BrowseTree example from the Forms Designer*

## Grid (8)

An fc\_Grid object makes use of the attributes packed into the ItemData property. The prefix property contains the number of columns. A grid object that contains row captions makes use of continuation items that are identified by a non-zero Medcinid property. Row captions are contained in the Description, Itemdata, and Prefix properties consecutively of the continuation item. More than one continuation item may be present if the captions exceed the combined allowable sizes of the property fields.

8 = grid/table layout

Columns: GUID 0

flags: right boundary of each column - i.e. width (e.g. "40|110|180|250|320|390|460|530|600" - 9 columns, the 1st 40 pixels wide and each subsequent one 70 pixels)

font: "|K=12171705|O=14474460|E=12500670|P=13|R=25|W=1"

caption: column labels (e.g. ""Ear|Outer Ear||Bulging|Erythema|Fluid|Mobility|Loss of Landmarks|Loss of Light Reflex")

Rows: GUID 1

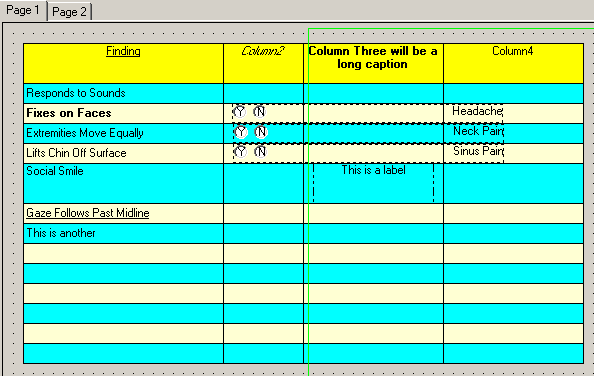
flags: ""

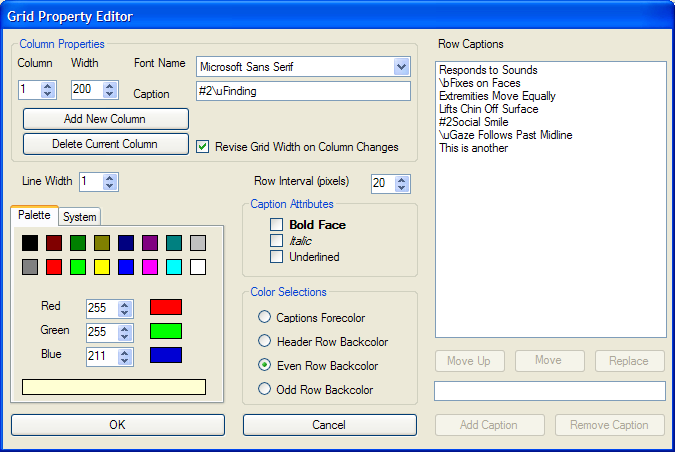
font: ""

caption: row labels (e.g. "Left|Right|||||||||||||||||||||")

It also looks like you can adjust the font inline on grid strings; the TSWF templates use Description strings like “\bJOINTS”; it’s possible that \i, \u, etc may also work. This doesn’t seem to work in other Description strings, though I didn’t heavily test it.

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| GridEvenRowBackColor | E=<int> | Specifies the back color of a Grid’s even rows |
| GridOddRowBackColor | O=<int> | Specifies the back color of a Grid’s odd rows |
| GridParentFrame | A=<int> | Specifies the parent frame associated with a Grid |
| GridParentRibbon | P=<int> | Specifies the parent ribbon of a Grid |
| GridRowInterval | R=<int> | Specifies the row interval of a Grid |
|  | W=<int> |  |

*Grid example from the Forms Designer*



*Grid property editor from the Forms Designer*

## Ribbon (16)

### ItemData

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| RibbonID | G=<int> | Specifies the ID of a Ribbon |
| RibbonLeftOffset | T=<int> | Specifies the distance (in pixels) of the left edge of a Ribbon from its button |
| RibbonRightOffset | R=<int> | Specifies the distance (in pixels) of the right edge of a Ribbon from its button’s LEFT |
| RibbonTopOffset | Y=<int> | Specifies the distance (in pixels) of the top edge of a Ribbon from its button |
| RibbonBottomOffset | B=<int> | Specifies the distance (in pixels) of the bottom edge of an expanded Ribbon from its button’s BOTTOM |
| RibbonCollapseOffset | C=<int> | Specifies the distance (in pixels) of the bottom edge of a collapsed Ribbon from its button |
| RibbonColumnSpan | S=<int> | Specifies the number of columns in a multicolumn page a Ribbon should span |
| RibbonChapterAssignment | H=<?> | Specifies the chapter assignment of a Ribbon |
| Options | O=<int> | Specifies additional options via a combined flag value (see below) |

|  |  |
| --- | --- |
| Value | Options Flags |
| 1 | expand Ribbon on yes |
| 2 | collapse Ribbon on no |
| 4 | hide Ribbon on collapse |
| 8 | show button |
| 16 | auto-expand Ribbon on no |
| 32 | auto-expand Ribbon on yes |

**Tested** **combinations**:

|  |  |  |
| --- | --- | --- |
| O=<Integer> | Expected behavior | Actual behavior |
| 3 | No button, expands on yes, collapses on no, collapses on empty | As expected |
| 9 | Has button, auto-expands on yes | Has button, no auto-behavior |
| 10 | Has button, auto-collapses on no | Has button, auto-collapses on yes |
| 11 | Has button, expands on yes, collapses on no, collapses on empty | As expected |
| 12 | Has button, hides ribbon on collapse | Ribbon starts in hidden state – cannot test |
| 24 | Has button, auto-expands on no | Has button, auto-expands on yes |
| 25 | Has button, auto-expands on yes OR no | Has button, auto-expands on yes |
| 26 | Has button, auto-expands on no, auto-collapse on no | Has button, auto-expands on yes, auto-collapses on yes |
| 40 | Has button, auto-collapses on yes | Has button, no auto-behavior |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Yes/No State** | **Expand on Yes** | **Expand on No** | **Collapse on Yes** | **Collapse on No** |
| **Yes Checked** | expand ribbon | collapse ribbon only if Collapse on Yes is set | collapse ribbon | n/a |
| **No Checked** | collapse ribbon only if Collapse on No is set | expand ribbon | n/a | collapse ribbon |
| **Unchecked** | collapse ribbon only if Collapse on No is set | collapse ribbon only if Collapse on Yes is set | n/a | n/a |

### Example

**001:** 1,10,397,37,419,0,16," |||||||0|0||0|0|||0|||0|0|0|0|0|0|||","N=3427|G=1|Y=5|B=51|C=5|O=24|R=1079|T=0","Domestic Violence Ribbon:"

**002:** 1,5,394,1075,469,0,2," |||||||0|0||0|0|||0|||0|0|1|0|0|0|||","N=3428|F=Arial|T=T|L=937:","986895:0:1:2:"

**003:** 1,40,400,745,419,0,32769," |||||||0|0||0|0|||0|||0|0|1|0|0|0|||","N=3429|F=Arial|B=T|T=T","Question inside the always-visible ribbon area"

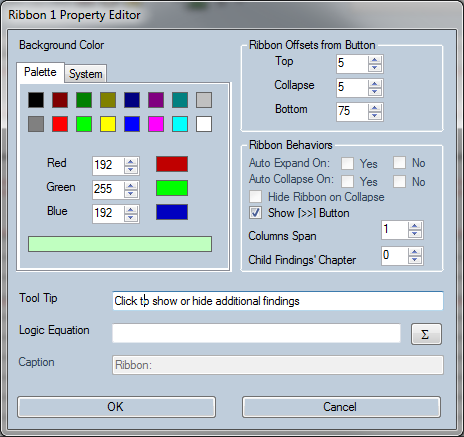
**004:** 1,745,400,808,420,115034,8705,"AO| || |TSWFPGEN|||19|63|YCN|0|0|X|X|0|||1|0|1|5|0|-1|||","N=3422|F=Arial|T=T|Y=6","Yes~Patient or parent IS currently in a situation where they are being verbally or physically hurt, threatened, or made to feel afraid."

**005:** 1,815,400,874,420,115034,8449,"AL| || |TSWFPGEN|||19|59|YCN|0|0|X|X|0|||0|0|1|5|0|-1|||","N=3421|F=Arial|T=T|Y=6","No~Patient or parent IS NOT currently in a situation where they are being verbally or physically hurt, threatened, or made to feel afraid."

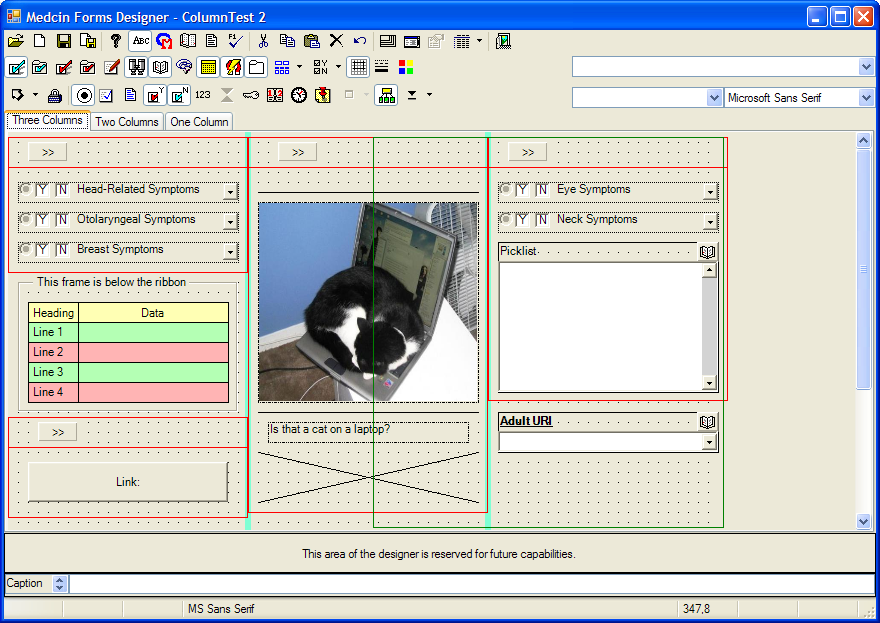
**006:** 1,5,423,1075,423,0,1," |||||||0|0||0|0|||0|||0|0|1|0|0|0|||","N=3424|F=Arial|L=","0:0:1:2:"

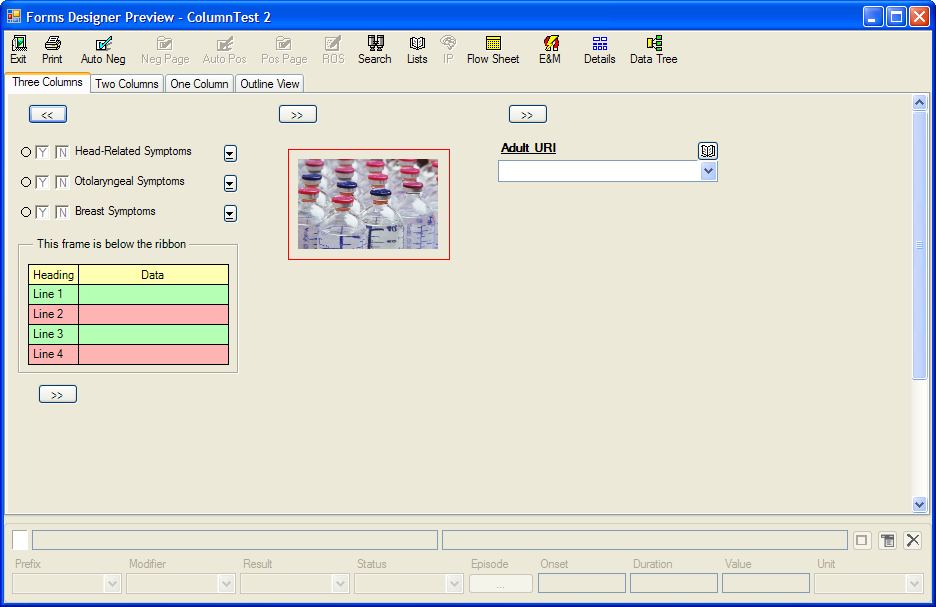
### Notes

* Specify object coordinates as if all ribbons are expanded (including items that come after and outside of the ribbon’s frame); they will shift upward automatically when the ribbon collapses
* Ribbons always consume the full width of the page, even if you specify narrower components. Visually a colored ribbon would not extend across the page, but functionally it does, so that the form knows how to shift other items up and down (even if the ribbon is invisible for an auto-neg button). For better ribbon “back color”, use colored frames inside the ribbon, leaving the ribbon itself transparent
* A ribbon will be on top of overlapping items that do not use it as a parent ribbon.
* To expand the ribbon in response to a checkbox being clicked, put the RibbonID in Prefix(17) of the triggering item



*Ribbon property editor from the Forms Designer*





*Example of column spanning using ribbons from the Forms Designer*

## Tabstrip (32)

The maximum number of pages supported is 26.

LINE 4/FLAG 32 (TABSTRIP): The first item other than the fc\_Form item may contain the backcolor of the form in the description property. When this occurs, the string starts with a colon followed by the numeric value of the color in decimal followed by a colon (e.g. “:12632256:Page 1|Page 2”). Each vertical bar delimited sub-string is the caption for a tab page. Each sub-string can be prefixed with a number sign, less than, or greater than character which designates no browsing, left lateral, or right lateral. This can be followed by a percent sign and numeric digit that indicates the number of columns specified for the page. When the page has a narrative chapter number assigned, it will appear as a suffix delimited with a tilde character.Form global data is also contained in the ItemData property of this item. This data is as follows:

The “Column Span” selector lets you edit the number of columns that the ribbon will span on a multi-column page (see the TabPage property editing for how to control the number of columns on a page).

|  |  |  |
| --- | --- | --- |
| FLAG | DESCRIPTION | VALUES |
| BS=<integer> | Specifies the border style of the page | 0 = border, 1 = no border |
| CB=<integer> | Specifies the default style of the yes and no entry component | 0 = full checkbox, 1 = terse checkbox, 2 = box-only checkbox, 4 = full radio button, 5 = terse radio button, 6 = box-only radio button |
| DF=<integer> | Specifies the availability of the entry details frame at runtime | 0 = unavailable, 1 = hidden but showable, 2 = visible but hideable |
| EM=<integer> | Specifies the availability of the E&M button | 0 = not available, 1 = available |
| FB=<integer> | Specifies the availability of the FlowSheet button | 0 = not available, 1 = available |
| HHL=<F> | *? not in original documentation, saw in FWST-Peds General* |  |
| MR=<char> | Specifies the multi-row property for TabStrip style page buttons | T or F |
| NB=<integer> | Specifies the availability of the auxiliary *negative* buttons | 0 = none, 1 = AutoNeg, , 2 = Neg all on page, 3 = both AutoNeg and Neg all on page |
| PB=<integer> | Specifies the availability of the auxiliary *positive* buttons | 0 = none, 1 = AutoPos, 2 = Pos all on page, 3 = both AutoPos and All pos on page |
| PL=<integer> | Specifies the availability of the PickList button | 0 = not available, 1 = available |
| PS=<integer> | *No longer used as the page button style is always TabStrip* |  |
| ROS=<integer> | Specifies the availability of the ROS button | 0 = not available, 1 = available |
| TP=<integer> | Specifies the placement of tabs for TabStrip style page buttons | 0 = top, 1 = bottom, 2 = left, 3 = right |
| TWS=<integer> | Specifies the tab width style for TabStrip style page buttons | 0 = justified, 1 = non-justified, 2 = fixed |
| V=<version string> | If present, this contains the version of the designer at the time the form object was saved. If not present, the version is less than 1 |  |

32 = tab bar; assigned tab should = template tab count (check)

dimensions: consistently 5,295 and 377,395 across 3 templates, yielding 290x18

67841 = checkbox (YCVU) - label - counterbox

caption: value range and label (e.g. ":0:10:1:Pain Level")

68353 = " ||||||min|19|80|YCUV|0|0|Y|N|0|||0|0|0|0|0||||" "" ":0:0:0:Loss of Consciousness from Head Trauma \_\_\_ Minutes"

76545 = " |||||||19|80|YCUVN|0|0|Y|N|0|||0|0|0|0|0||||" "" ":0:0:0:Mass Palpated"

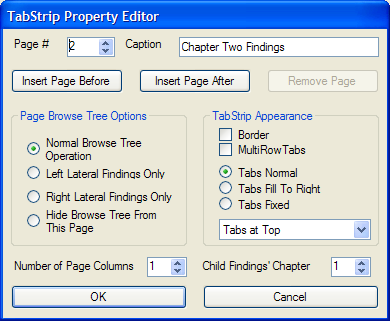
" ||||||cm|19|80|YCUVN|0|0|Y|N|0|||0|0|0|0|0||||" "|T=T|L=" ":0:0:0:Mass Present"

Maximum number of 26 pages is supported.

The more controls that are used in your form design, the greater the amount of system resources needed to show your form at runtime. Performance also degrades as the quantity of form components increases. You may want to consider breaking large forms into multiple forms with link buttons to navigate the form set.

You can assign all findings that are on a form page to a particular narrative chapter by setting the Child Findings’ Chapter selector to a value other than 0. In the designer, this assignment will display as part of the caption delimited with a tilde character; it will not appear at run time. Note that the assignment, if any, of form findings within a frame or ribbon will take precedent over the page assignment. Chapter numbers in the forms designer are 0-based. This means that when chapter assignments are not made, all findings are placed in chapter 0. When this occurs, the Medcin narrative engine does not display a chapter header. Chapter numbers can be assigned from 0 to 100. When the narrative engine emits a numbered header for a chapter, the number displayed is one greater than the assigned number so that chapter 0 is displayed as chapter 1. Keep this in mind when assigning your chapter numbers. Evidence of this can be seen in the screen capture of the dialog above; the assignment is 1 but the caption properly reflects this as Chapter Two Findings.

* **Justified**: Each tab is wide enough to accommodate its contents and, if needed, the width of each tab is increased so that each row of tabs spans the width of the control. If the *Multiple**Rows* menu item is unchecked, or if there is only a single row of tabs, this setting has no effect. This corresponds to a DotNet setting of “*Normal*”.
* **Non-justified**: Each tab is just wide enough to accommodate its contents. The rows are not justified, so multiple rows of tabs are jagged. This corresponds to a DotNet setting of “*Fill to right*”.
* **Fixed**: All tabs have an identical width, which is determined by the width of the longest caption.



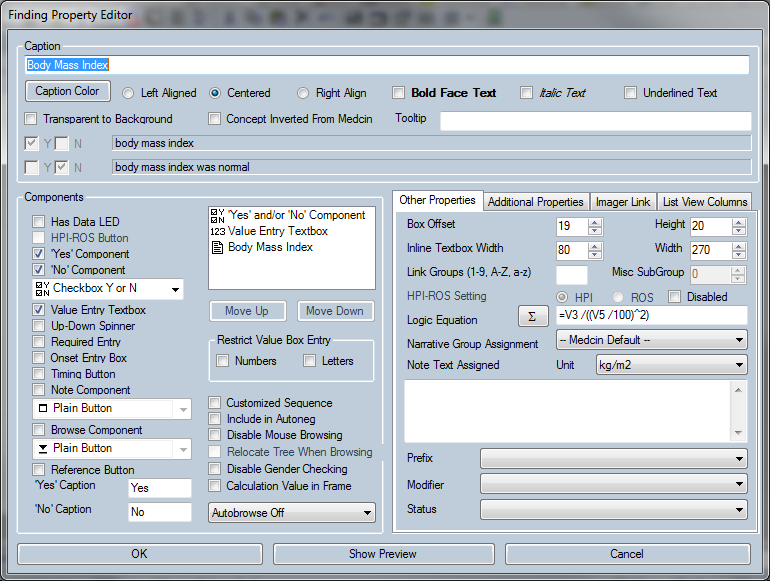
*TabStrip property editor from the Forms Designer*

## Medcin Finding object

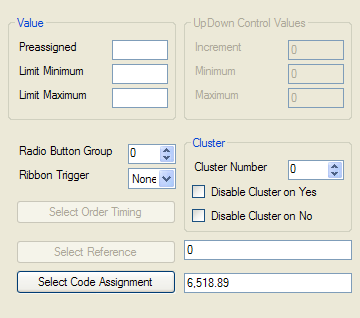
Items designed to be attached to a Medcin object may not be visible unless there is a valid number assigned.

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| Options | O=<int> | Specifies additional options via a combined flag value (see below) |

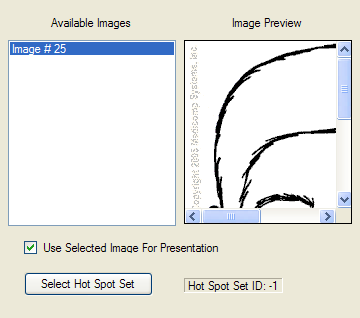
|  |  |
| --- | --- |
| Value | Options Flags |
| 1 | Medcin object: Used internally for “Finding description has been edited” |
| 2 | Medcin object: Disable mouse browsing for this finding |
| 8 | Medcin object:: Disable cluster on Y |
| 16 | Medcin object: Disable cluster on N |
| 256 | Designer inverted concept from Medcin |



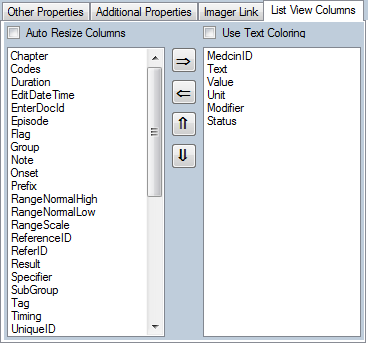
*Medcin finding property editor from the Forms Designer*



*Additional Properties tab*



*Imager Link tab*



*List View Columns tab*

## Checkbox (256, 512)

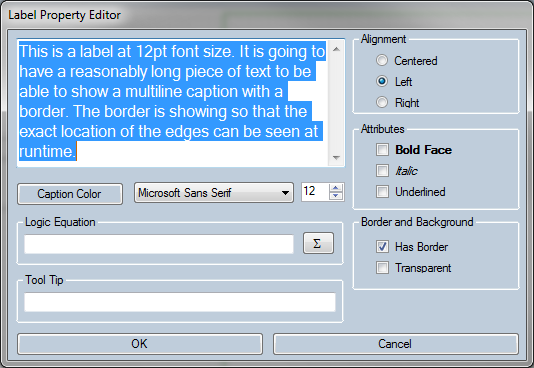
enum CheckboxStyle { Full = 0, Terse = 1, BoxOnly = 2, FullRadiobutton = 4, TerseRadiobutton = 5, BoxOnlyRadiobutton = 6 };

A “Yes” box chkyes is a component available to a Medcin finding object; its caption is set to one of three states based on the current setting of the “Yes” toolbar button menu: “Yes”, “Y”, or simply a box without a caption and an interior state indicator. When in a pressed state, added or edited Medcin objects will include a box for indicating “Yes” or positive. The box can also be styled as a radio button; however, for true radio button operation, you will need to make the finding part of a radio button group. You also can change the caption using the property dialog. At runtime, checking the yes component has the same result as selecting the PLUS button of the data entry tree; the finding is entered as positive. You can design an opposite action by means of the “Concept Inverted From Medcin” property discussed in the section on the object’s property dialog. See prefix #19 above for an alternate way of doing exclusive selectivity (i.e. other than radiobuttons).

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| CheckboxYesNoStyle | Y=<0|1|2>  Y=<4|5|6> | Specifies the appearance of Checkboxes  Specifies the appearance of RadioButtons |

## Label (32768)

An fc\_Label object makes use of the ItemData property attributes. Alignment is interpreted based on the presence of the fc\_Left or fc\_Right flag property bits. The caption interprets the presence of single tilde characters (“~”) in the description property as a CR-LF pair.



*Label property editor from the Forms Designer*

## Reference (262144)

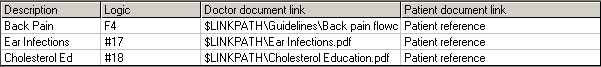
For a fc\_Reference object, the caption is contained in the Description property. The ItemData property contains the Logic, Doctor reference path, and Patient reference path delimited with a Tab character (e.g. “Logic[TAB]DRP[TAB]PRP”). You can discriminate a reference object from a Medcin object since a reference object will have only the fc\_Reference bit set. At this time, only the doctor document link is implemented.

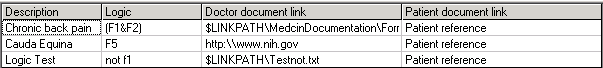
262144 = URLs; listed at end of document

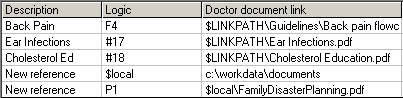
flags: left blank

font: #<refnum>[TAB]<URL>[TAB]<label vs hover text> (no spaces!)

caption: label vs hover text?







*Examples of Form Reference Documents from the Forms Designer*

Tested options:

No ItemData flags (including O) seem to have no effect on the appearance of the ? button, although font size can still be affected with Z

Component order: YCR = text followed by ? button; RCY = ? button followed by text

## Link (16384)

An fc\_Link object is interpreted as a button that is associated with another form or a reference. The description property contains the button’s caption and the name of the associated form or reference delimited with a colon. If the button has a pre-assigned message, the message string will be appended to the end of the description property delimited with a tilde (“~”) character. The MedcinId field contains the FormID of the associated form. On a multi-page form, if the Page property is 0, the button will appear on all form pages, otherwise it will appear only on the designated page. A FormSID value, if available, is in the 1st Prefix sub-field.

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| Options | O=<int> | Specifies additional options via a combined flag value (see below) |

|  |  |
| --- | --- |
| Value | Options Flags |
| 1 | Link button object: Fire when first logic True |
| 2 | Link button object: Raise message when first logic True |
| 4 | Link button: Button is a user button |
| 8 | Link button object: Button is an autoneg action button |
| 16 | Link button object: Button is an autopos action button |
| 32 | Link Button object: Button is a negblock action button |
| 64 | Link Button object: Button is a posblock action button |
| 128 | Link Button object: Button is a flowsheet action button |
| 256 | Link Button object: Button is a reference button |

Example of linking to a URL:

12,10,401,430,426,0,16384," |||||||0|0||0|0|||0|||0|0|0|0|0|0|||","N=2983|O=257","RelayHealth/MiCare/Secure Messaging:#63"

0,0,0,0,0,0,262144," ","#63 https://app.relayhealth.com/welcome.aspx Patient reference","Relay Health"

### Autonegative buttons

**Requires**:

* Ribbon object (parent ribbon): frame and all findings must have this as parent ribbon
* Frame object (enclosing frame): must enclose all objects affected by autonegative (including the autoneg button itself); *each must have a different logic identifier if you want to use them with the same ribbon* (i.e.L=1:, L=2:, etc). Unclear if it *must* be contained within the parent ribbon
* Link object (triggering button): caption should convey the behavior to the user
* Items eligible for Autonegative (Y/N behavior)

Action buttons do not support a logic equation and do not support a message.

A button’s parent block is interpreted as follows:

* If the button is within a frame component, then all of the form findings that are within the frame constitute the parent block.
* If a button is within a ribbon and not within a frame, then all of the findings that are not in a frame that are within the ribbon constitute the parent block.
* If a button is neither in a frame nor in a ribbon, then all of the form findings that are on the form page but are not in frames or ribbons constitute the parent block.

An action button can be designed to invoke one of the following actions on the parent block *of findings previously documented\**:

* **Auto-Negative** (8): automatic negative functionality for symptom and physical Medcin objects in the parent block;
* **Auto-Positive** (16): automatic positive functionality for therapy Medcin objects in the parent block.
* **Make Negative** (32): negate all the form findings in the parent block that have a “No” checkbox for regular findings, or a “Yes” checkbox for inverse normal findings
* **Make Positive** (64): make all the form findings positive in the parent block that have a “Yes” checkbox for regular findings, or a “No” checkbox for inverse normal findings.
* **Flow Sheet** (128): the “flow sheet of a form page” functionality will be invoked but limited to the findings within the parent block.

\* *This is ambiguous – does it account for the behavior of previously checked items re-checking themselves? This is seen when a bunch of items in the parent block are checked and saved; then unchecked and saved; exit the patient completely, and then re-enter the encounter, and the items are re-checked (at least when using O=64/autopositive)*

*It’s also worth noting the specific use of “symptom and physical Medcin objects” – in a test, the chief complaint finding (1718) was excluded.*

**Designer wording:**

Auto-Positive empty findings in the parent block

Make empty findings in the parent block negative

Make empty findings in the parent block positive

Flow Sheet the findings in the parent block

Delete entered findings in the parent block *(“Obsolete Terms” tab in TSWF uses O=512, so clearly some options flags are missing)*

*Options found in the wild*

Example:

1,10,33,31,53,0,16," |||||||0|0||0|0|||0|||0|0|0|0|0|0|||","N=3439|G=1|Y=5|B=927|C=6|O=0|R=1079|T=0","ROS Ribbon:"

1,5,419,219,609,0,2," |||||||0|0||0|0|||0|||0|0|1|0|0|0|||","N=3514|F=Arial|T=T","986895:0:1:1:"

1,10,424,214,445,0,16384," |||||||0|0||0|0|||0|||0|0|1|0|0|0|||","N=3515|F=Arial|B=T|O=32","AutoNeg CONSTITUTIONAL:"

1,10,449,214,469,7,268445441," | || ||||19|80|YCHN|0|0|Y|N|0|||0|0|1|0|0|0|||","N=3516|F=Arial|T=T","Chills"

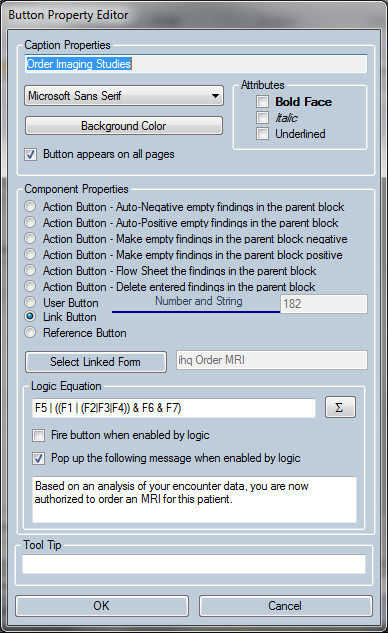
1,10,469,214,489,272,268445441," |||||||19|80|YCHN|0|0|Y|N|0|||0|0|1|0|0|0|||","N=3517|F=Arial|T=T","Night Sweats"

1,10,489,214,509,115277,268445441," | || ||Z||19|80|YCHN|0|0|Y|N|0|||0|37|1|0|0|0|||","N=3518|O=25|F=Arial|T=T","Feeling Tired"

1,10,509,214,529,1002,268445441," | || ||||19|80|YCHN|0|0|Y|N|0|||0|0|1|0|0|0|||","N=3519|F=Arial|T=T","Muscle Aches"

1,10,564,214,584,281576,268445441," | || ||||19|80|YCHN|0|0|Y|N|0|||0|0|1|0|0|0|||","N=3522|F=Arial|T=T","Abnormal Head Shape"

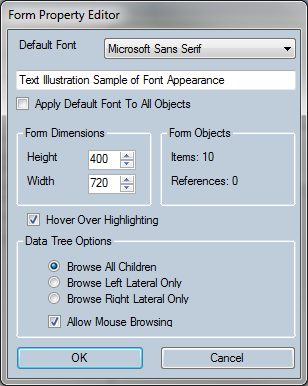
1,10,584,214,604,90,268445441," |||||||19|80|YCHN|0|0|Y|N|0|||0|0|1|0|0|0|||","N=3523|F=Arial|T=T","Sinus Pain"



*Button property editor from the Forms Designer*

## Form (1048576)

For the fc\_Form component, PositionRight contains the form’s width and PositionBottom contains the form’s height. For all other presentation objects, the position properties describe the bounding rectangle of the object.



*Form property editor from the Forms Designer*

## Lists (PickList, List, ListView)

For an fc\_Picklist object, the Medcinid property contains the list’s ListID. The description property contains the caption of the list and the list name delimited by a colon (e.g. “If you have lower back pain:LOWER BACK PAIN”). The picklist object uses the link group and box offset sub-fields of the prefix property. It may make use of the attributes packed into the ItemData property. When it is an internal drop list, the list of Medcinids is contained in the ItemData field. A ListSID value, if available, is in the 1st Prefix sub-field.

Only internal lists are supported in the current version of AHLTA.

Comma-delimited list of MedcinIds

ListID

Example (dropdown):

1,10,85,300,125,**16**,159383553,

" *[ListSID]* ||||| *[LinkGroup]* || *[BoxOffset]* 19 |0||0|0|||0|||0|0|0|0|0|0|||",

" B=T|L=123467,606,111296,607,608,123324,609,128024,2115,111297,129250,129251,122413,122414,605",

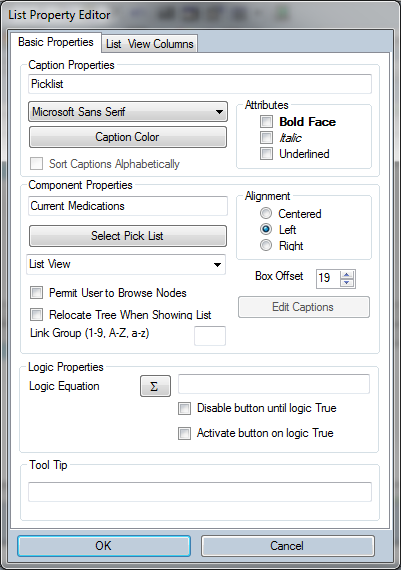
"Birth Control Method:BirthControl"

List name

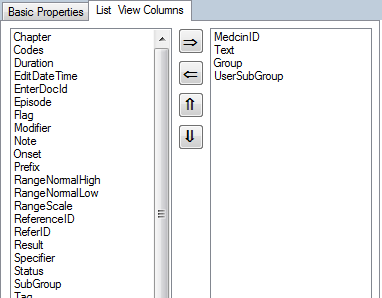
List caption

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| ListViewColumnItems | G=<?> | Specifies the comma-delimited list of chart column item enum values for a ListView associated with Medcin or Picklist objects |
| Options | O=<int> | Specifies additional options via a combined flag value (see below) |

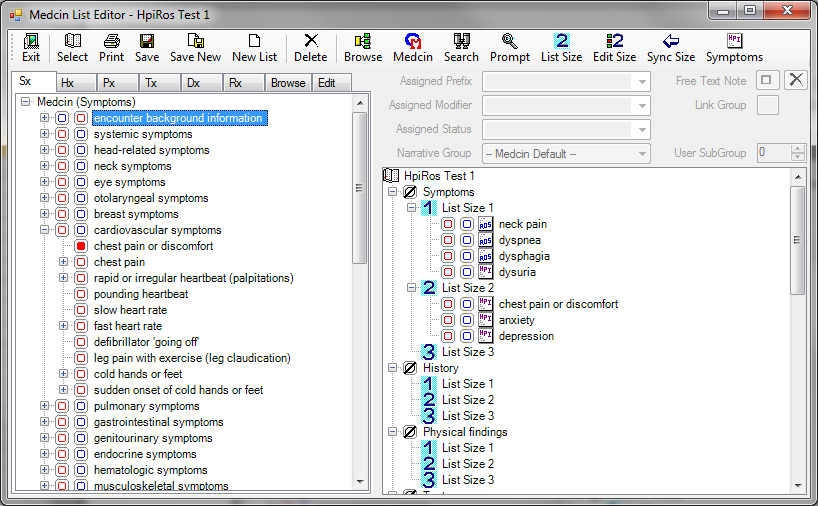
|  |  |
| --- | --- |
| Value | Options Flags |
| 2 | List Object: Allow node expansion |
| 4 | List Object: Logic state; disabled=0, enabled=1 |
| 8 | List Object: Disable until logic true |
| 16 | List Object: Activate on first logic true |
| 32 | List Object: Relocate origin of treeview relative to object |
| 64 | List Object: Sort inline list captions |
| 128 | ListView Object: Use color |
| 156 | ListView Object: auto resize columns |



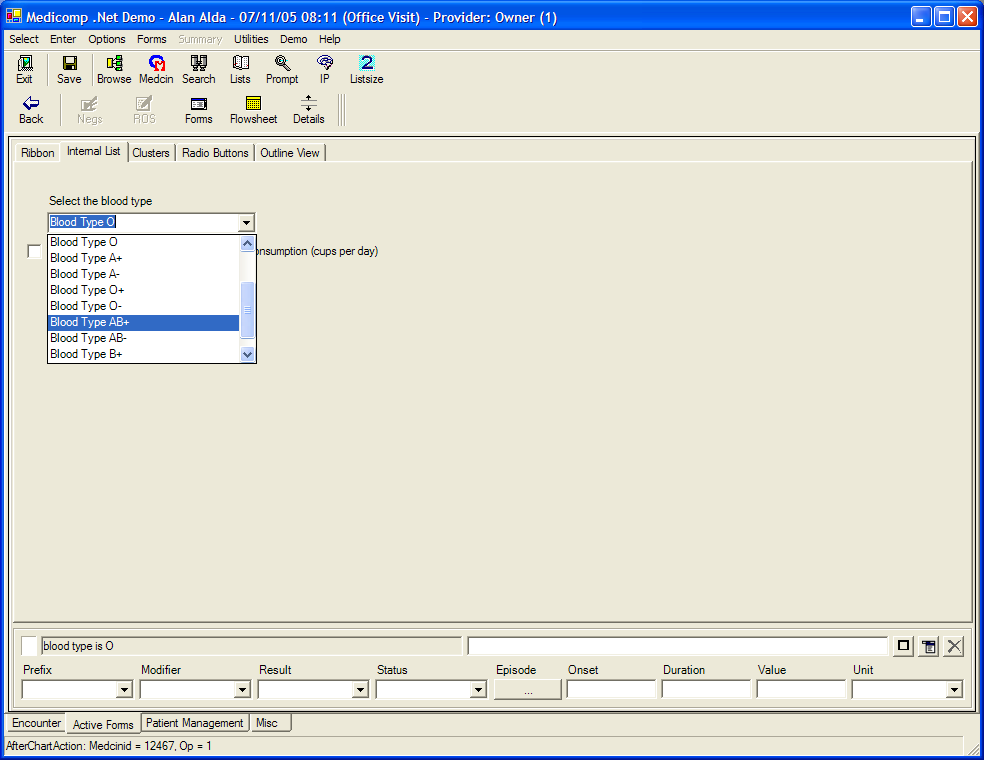
*List property editor from the Forms Designer*



*List View Columns tab*

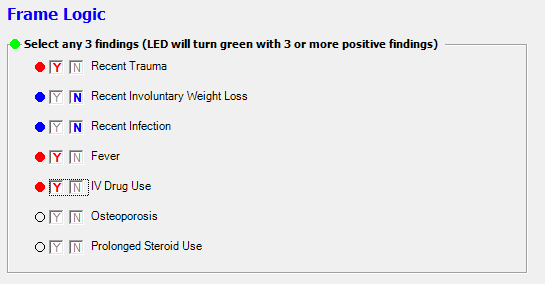


*List editor from the Forms Designer*

**

*Example of a drop-down list (essentially a short picklist contained within the dropdown portion of a component) from the Forms Designer*

## LED / “HasData” (131072)



3,10,50,530,280,0,131074," |||||||0|0||0|0|||0|||0|0|0|0|0|0|||","N=32|F=Tahoma|B=T|L=1:G1","0:0:1:1:Select any 3 findings (LED will turn green with 3 or more positive findings)"

3,35,65,450,85,3438,131841," |RE|| ||||19|80|LYC|0|0|||0|||0|0|0|0|0|-1|1||","N=34|T=T|Y=2|L=1(3)","Recent Trauma"

3,35,95,450,115,2993,131841," | || |||lbs|19|80|LYC|0|0|||0|||0|0|0|0|0|-1|1||","N=35|T=T|Y=2|L=1","Recent Involuntary Weight Loss"

3,35,125,450,145,3479,131841," | || ||||19|80|LYC|0|0|||0|||0|0|0|0|0|-1|1||","N=36|T=T|Y=2|L=1","Recent Infection"

## Note (8192)

An fc\_Note object with a Medcinid of 0 is used to contain string information as follows: For Page=0, the chapter heading string is split across the description and itemdata properties. For Page=1, the SubGroup heading string is split across the description and itemdata properties.

## UpDown (65536)

An fc\_Medcin object makes use of the packed prefix property and the packed ItemData property. If a free text string is pre-assigned, it will be appended to the description property delimited with a tilde (“~”) character. If the object incorporates an UpDown component, the minimum, maximum, and increment values are packed into the front of the description property delimited with colons and starting with a colon (e.g. “:0:10:1:daily coffee consumption”).

Uses the value, min, max fields

## ValueBox (2048)

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| ValueBoxChars | V=<0|1|2> | Specifies what type of characters are allowed in a ValueBox:  AllowAll (0), AllowNumbers (1), AllowLetters (2) |

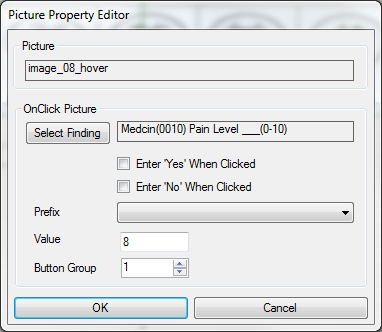
## Picture (64)

**This Medcin feature is not implemented in the current version of AHLTA.**

For an fc\_Picture object, the description property is expected to be the index key of the picture in the FormPictures collection. If the object is an imager control, the ItemData property is used for additional properties, and the description is empty.

|  |  |  |
| --- | --- | --- |
| Option | Flag | Description |
| ImageMarkerStyle | M=<int> | Specifies the composite image marker style |
| Options | O=<int> | Specifies additional options via a combined flag value (see below) |

|  |  |
| --- | --- |
| Value | Options Flags |
| 2 | Picture object: Picture is an imager component |
| 4 | Picture object: Set associated finding positive on click |
| 8 | Picture object: Set associated finding negative on click |
| 16 | Picture object: Set associated finding’s prefix on click |
| 32 | Picture object: Set associated finding’s value on click |



*Picture property editor from the Forms Designer*

## Window (536870912)

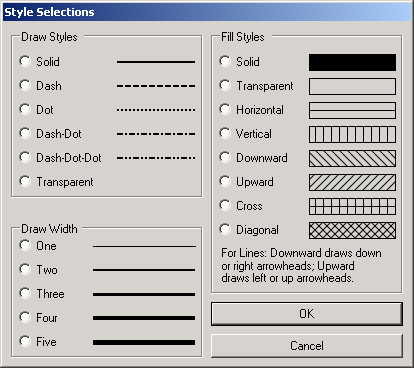
**This Medcin feature is not implemented in the current version of AHLTA.**

For an fc\_Window object, the caption and signature are packed into the description property delimited with a colon (e.g. “Sample Demographics:MedDemo.Demog”). Window objects are not supported in AHLTA.

# Styles

|  |  |
| --- | --- |
| Bits | Description |
| 0-2 | Fill Style |
| 3-5 | Draw Width |
| 6-8 | Draw Style |
| 9-10 | Size: 0=full, 1=3/4, 2=1/2, 3=1/4 |
| 11 | Shape: 0=rectangular, 1=ovoid |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Outline Style | Flag | Example | Fill Style | Flag | Example | Fill Style | Flag | Example |
| Solid | 0 |  | Solid | 0 |  | UpwardDiagonal | 4 |  |
| Dash | 1 |  | Transparent | 1 |  | DownwardDiagonal | 5 |  |
| Dot | 2 |  | HorizontalLine | 2 |  | Cross | 6 |  |
| DashDot | 3 |  | VerticalLine | 3 |  | DiagonalCross | 7 |  |



*Style Selections dialog from the Forms Designer*

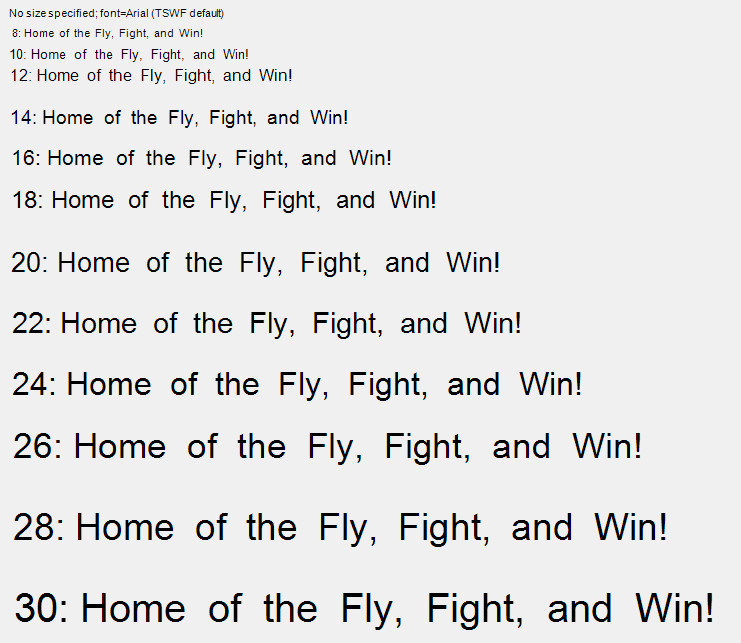
# Fonts

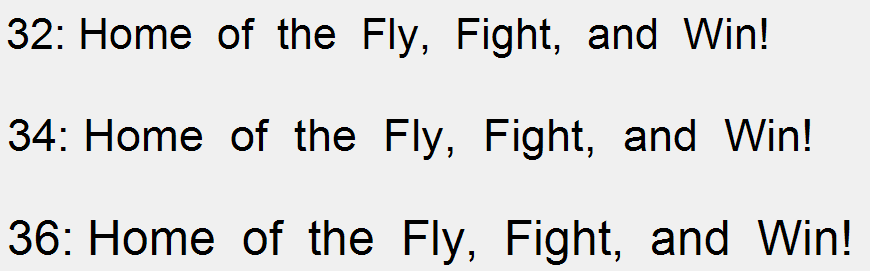
## Tested fonts

|  |  |  |
| --- | --- | --- |
| Font | String | Example (at Z=14) |
|  | F=Arial |  |
|  | F=Broadway |  |
|  | F=Calibri |  |
|  | F=Century |  |
|  | F=Consolas |  |
|  | F=Elephant |  |
|  | F=Garamond |  |
|  | F=Georgia |  |
|  | *default* |  |
|  | F=Perpetua |  |
|  | F=Rockwell |  |
|  | F=Stencil |  |
|  | F=Tahoma |  |
|  | F=Verdana |  |
|  | F=Wingdings |  |

**Not working**: ArialRounded, Bauhaus, Bauhaus93, Bauhaus93Reg, Bauhaus93Regular, BookmanAntiqua, BookmanOldStyle, CenturyRegular, CenturyGothic, Cooper, CooperBlack, Courier, CourierNew, CourierRegular, FelixTitling, FelixTitlingRegular, StencilRegular, Trebuchet, TrebuchetMS

## Font size (Z=<int>)





## Font tricks

Bullet, hollow square: "F=Wingdings|T=T", "o"

Bullet, solid square: "F=Wingdings|T=T", "n"

## Notes

Regardless of font properties set for the template, the outputted font in the “Current Encounter” section of the note is Arial, 8-point.

# Colors

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | AHLTA INT | COLOR NAME | HEX | RGB |
|  | 16118015 | LavenderBlush | FFF0F5 |  |
|  | 12695295 | LIghtPink | FFB6C1 |  |
|  | 13353215 | Pink | FFC0CB |  |
|  | 9639167 | DeepPink | FF1493 |  |
|  | 11823615 | HotPink | FF69B4 |  |
|  | 14204888 | Thistle | D8BFD8 |  |
|  | 14524637 | Plum | DDA0DD |  |
|  | 15631086 | Violet | EE82EE |  |
|  | 14053594 | Orchid | DA70D6 |  |
|  | 9662683 | PaleVioletRed | DB7093 |  |
|  | 8721863 | MediumVioletRed | C71585 |  |
|  | 3937500 | Crimson | DC143C |  |
|  | 16711935 | Magenta | FF00FF |  |
|  | 9109643 | DarkMagenta | 8B008B |  |
|  | 13850042 | MediumOrchid | BA55D3 |  |
|  | 13382297 | DarkOrchid | 9932CC |  |
|  | 13458026 | SlateBlue | 6A5ACD |  |
|  | 15624315 | MediumSlateBlue | 7B68EE |  |
|  | 9125192 | DarkSlateBlue | 483D8B |  |
|  | 14381203 | MediumPurple | 9370DB |  |
|  | 8388736 | Purple | 800080 |  |
|  | 14822282 | BlueViolet | 8A2BE2 |  |
|  | 13828244 | DarkViolet | 9400D3 |  |
|  | 8519755 | Indigo | 4B0082 |  |
|  | 16775416 | GhostWhite | F8F8FF |  |
|  | 16775408 | AliceBlue | F0F8FF |  |
|  | 16777200 | Azure | F0FFFF |  |
|  | 16777184 | LightCyan | E0FFFF |  |
|  | 16443110 | Lavender | E6E6FA |  |
|  | 15658671 | PaleTurquoise | AFEEEE |  |
|  | 15130800 | PowderBlue | B0E0E6 |  |
|  | 15128749 | LightBlue | ADD8E6 |  |
|  | 14599344 | LightSteelBlue | B0C4DE |  |
|  | 16436871 | LightSkyBlue | 87CEFA |  |
|  | 15453831 | SkyBlue | 87CEEB |  |
|  | 10061943 | LightSlateGray | 778899 |  |
|  | 9470064 | SlateGray | 708090 |  |
|  | 5197615 | DarkSlateGray | 2F4F4F |  |
|  | 10526303 | CadetBlue | 5F9EA0 |  |
|  | 11829830 | SteelBlue | 4682B4 |  |
|  | 15570276 | CornFlowerBlue | 6495ED |  |
|  | 16748574 | DodgerBlue | 1E90FF |  |
|  | 14772545 | RoyalBlue | 4169E1 |  |
|  | 13434880 | MediumBlue | 0000CD |  |
|  | 16711680 | Blue | 0000FF |  |
|  | 9109504 | DarkBlue | 00008B |  |
|  | 8388608 | Navy | 000080 |  |
|  | 7346457 | MidnightBlue | 191970 |  |
|  | 16760576 | DeepSkyBlue | 00BFFF |  |
|  | 13749760 | DarkTurquoise | 00CED1 |  |
|  | 16776960 | Cyan | 00FFFF |  |
|  | 9145088 | DarkCyan | 008B8B |  |
|  | 8421376 | Teal | 008080 |  |
|  | 16449525 | MintCream | F5FFFA |  |
|  | 13959039 | Aquamarine | 7FFFD4 |  |
|  | 11193702 | MediumAquamarine | 66CDAA |  |
|  | 13422920 | MediumTurquoise | 48D1CC |  |
|  | 13688896 | Turquoise | 40E0D0 |  |
|  | 7451452 | MediumSeaGreen | 3CB371 |  |
|  | 5737262 | SeaGreen | 2E8B57 |  |
|  | 11186720 | LightSeaGreen | 20B2AA |  |
|  | 10156544 | MediumSpringGreen | 00FA9A |  |
|  | 8388352 | SpringGreen | 00FF7F |  |
|  | 15794160 | HoneyDew | F0FFF0 |  |
|  | 10025880 | PaleGreen | 98FB98 |  |
|  | 9498256 | LightGreen | 90EE90 |  |
|  | 9419919 | DarkSeaGreen | 8FBC8F |  |
|  | 3329330 | LimeGreen | 32CD32 |  |
|  | 3329434 | YellowGreen | 9ACD32 |  |
|  | 3145645 | GreenYellow | ADFF2F |  |
|  | 2330219 | OliveDrab | 6B8E23 |  |
|  | 3107669 | DarkOliveGreen | 556B2F |  |
|  | 2263842 | ForestGreen | 228B22 |  |
|  | 32768 | Green | 008000 |  |
|  | 25600 | DarkGreen | 006400 |  |
|  | 65280 | Lime | 00FF00 |  |
|  | 64636 | LawnGreen | 7CFC00 |  |
|  | 65407 | Chartreuse | 7FFF00 |  |
|  | 16448255 | Snow | FFFAFA |  |
|  | 15794175 | Ivory | FFFFF0 |  |
|  | 15792895 | FloralWhite | FFFAF0 |  |
|  | 15660543 | Seashell | FFF5EE |  |
|  | 15136253 | OldLace | FDF5E6 |  |
|  | 15134970 | Linen | FAF0E6 |  |
|  | 14804223 | MistyRose | FFE4E1 |  |
|  | 14745599 | LIghtYellow | FFFFE0 |  |
|  | 14481663 | CornSilk | FFF8DC |  |
|  | 14480885 | Beige | F5F5DC |  |
|  | 14150650 | AntiqueWhite | FAEBD7 |  |
|  | 14020607 | PapayaWhip | FFEFD5 |  |
|  | 13826810 | LightGoldenrodYellow | FAFAD2 |  |
|  | 13499135 | LemonChiffon | FFFACD |  |
|  | 13495295 | BlanchedAlmond | FFEBCD |  |
|  | 12903679 | Bisque | FFE4C4 |  |
|  | 12180223 | PeachPuff | FFDAB9 |  |
|  | 11920639 | Moccasin | FFE4B5 |  |
|  | 11788021 | Wheat | F5DEB3 |  |
|  | 11394815 | NavajoWhite | FFDEAD |  |
|  | 11200750 | PaleGoldenrod | EEE8AA |  |
|  | 9408444 | RosyBrown | BC8F8F |  |
|  | 9234160 | Khaki | F0E68C |  |
|  | 9221330 | Tan | D2B48C |  |
|  | 8894686 | Burlywood | DEB887 |  |
|  | 8421616 | LightCoral | F08080 |  |
|  | 8036607 | LightSalmon | FFA07A |  |
|  | 8034025 | DarkSalmon | E9967A |  |
|  | 7504122 | Salmon | FA8072 |  |
|  | 7059389 | DarkKhaki | BDB76B |  |
|  | 6333684 | SandyBrown | F4A460 |  |
|  | 5275647 | Coral | FF7F50 |  |
|  | 4678655 | Tomato | FF6347 |  |
|  | 4163021 | Peru | CD853F |  |
|  | 6053069 | IndianRed | CD5C5C |  |
|  | 2970272 | Sienna | A0522D |  |
|  | 2763429 | RedBrown | A52A2A |  |
|  | 2237106 | FireBrick | B22222 |  |
|  | 2139610 | Goldenrod | DAA520 |  |
|  | 1993170 | Chocolate | D2691E |  |
|  | 1262987 | SaddleBrown | 8B4513 |  |
|  | 755384 | DarkGoldenrod | B8860B |  |
|  | 65535 | Yellow | FFFF00 |  |
|  | 32896 | Olive | 808000 |  |
|  | 55295 | Gold | FFD700 |  |
|  | 42495 | Orange | FFA500 |  |
|  | 36095 | DarkOrange | FF8C00 |  |
|  | 17919 | OrangeRed | FF4500 |  |
|  | 255 | Red | FF0000 |  |
|  | 139 | DarkRed | 8B0000 |  |
|  | 128 | Maroon | 800000 |  |
|  | 16777215 | White | FFFFFF |  |
|  | 16119285 | WhiteSmoke | F5F5F5 |  |
|  | 14474460 | Gainsboro | DCDCDC |  |
|  | 13882323 | LightGray | D3D3D3 |  |
|  | 12632256 | Silver | C0C0C0 |  |
|  | 11119017 | DarkGray | A9A9A9 |  |
|  | 8421504 | Gray | 808080 |  |
|  | 908265 | DimGray | 696969 |  |
|  | 0 | Black | 000000 |  |

## Narrative Group Number

GroupNumber is one of the following constants exposed by the control:

|  |  |  |
| --- | --- | --- |
| See text | medAllGroups |  |
| Allergies | medAllergies | 24 |
| Assessment | medAssessment | 11 |
| ChiefComplaint | medChiefComplaint | 1 |
| CounselingEducation | medCounselingEducation | 22 |
| CurrentMedication | medCurrentMedication | 3 |
| Discussed | medDiscussed | 25 |
| FamilyHistory | medFamilyHistory | 7 |
| History | medHistory | 6 |
| HistoryOfPresentIllness | medHistoryPresentIllness | 2 |
| Misc1 | medGroupMisc1 | 28 |
| Misc2 | medGroupMisc2 | 29 |
| Misc3 | medGroupMisc3 | 30 |
| Misc4 | medGroupMisc4 | 31 |
| Misc5 | medGroupMisc5 | 32 |
| Misc6 | medGroupMisc6 | 33 |
| Misc7 | medGroupMisc7 | 34 |
| Misc8 | medGroupMisc8 | 35 |
| Misc9 | medGroupMisc9 | 36 |
| Misc10 through Misc37 | MedGroupMisc10, etc. | 37 through 64 |
| Notes | medNotes | 14 |
| Objective | medObjective | 9 |
| Other | medOther | 15 |
| PastMedicalHistory | medPastMedicalHistory | 4 |
| PersonalHistory | medPersonalHistory | 5 |
| PhysicalFindings | medPhysicalFindings | 8 |
| Plan | medPlan | 13 |
| PracticeManagement | medPracticeManagement | 17 |
| PreviousTests | medPreviousTests | 18 |
| PreviousTherapy | medPreviousTherapy | 19 |
| Reason for visit | medReasonForVisit | 27 |
| ReviewOfSystems | medReviewOfSystems | 16 |
| ReferredHere | medReferredHere | 20 |
| TestConclusions | medTestConclusions | 21 |
| Tests | medTests | 10 |
| Therapy | medTherapy | 12 |
| RuleOut | medRuleOut | 26 |
| Vaccinations | medVaccinations | 23 |

## Narrative Group Sequence

The initial sequence of groups for presentation purposes is as follows (order is global define, name, exposed constant):

|  |  |  |
| --- | --- | --- |
| G\_REASONVISIT | Reason for visit | medReasonForVisit |
| G\_REASON | Chief Complaint | medChiefComplaint |
| G\_REFERDX | Referred Here | medReferredHere |
| G\_HPI | History Of Present Illness | medHistoryPresentIllness |
| G\_CURMED | Current Medication | medCurrentMedication |
| G\_PAST\_MED\_HIST | Past Medical History | medPastMedicalHistory |
| G\_PAST\_THERAPY | Previous Therapy | medPreviousTherapy |
| G\_PER\_HIST | Personal History | medPersonalHistory |
| G\_HIST | History | medHistory |
| G\_FAM\_HIST | Family History | medFamilyHistory |
| G\_ROS | Review Of Systems | medReviewOfSystems |
| G\_PHYSICAL | Physical Findings | medPhysicalFindings |
| G\_OBSERV | Objective | medObjective |
| G\_TESTS | Tests | medTests |
| G\_RULEOUT | Rule Out | medRuleOut |
| G\_PAST\_TESTS | Previous Tests | medPreviousTests |
| G\_ASSESS | Assessment | medAssessment |
| G\_TESTCON | Test Conclusions | medTestConclusions |
| G\_THERAPY | Therapy | medTherapy |
| G\_ALLERGYS | Allergies | medAllergies |
| G\_VACCINES | Vaccinations | medVaccinations |
| G\_EDUCATION | Counseling and Education | medCounselingEducation |
| G\_DISCUSS | Discussed | medDiscussed |
| G\_PLAN | Plan | medPlan |
| G\_NOTE | Notes | medNotes |
| G\_OTHER | Other | medOther |
| G\_EVAL | Practice Management | medPracticeManagement |
| G\_MISC1 | Misc1 | medGroupMisc1 |
| G\_MISC2 | Misc2 | medGroupMisc2 |
| G\_MISC3 | Misc3 | medGroupMisc3 |
| G\_MISC4 | Misc4 | medGroupMisc4 |
| G\_MISC5 | Misc5 | medGroupMisc5 |
| G\_MISC6 | Misc6 | medGroupMisc6 |
| G\_MISC7 | Misc7 | medGroupMisc7 |
| G\_MISC8 | Misc8 | medGroupMisc8 |
| G\_MISC9 – G\_MISC37 | Misc9 – Misc37 | medGroupMisc9 – medGroupMisc37 |

## String length

1. Length of strings assigned by the Form’s Designer to the Description, ItemData, and Prefix properties will be limited to the field sizes for these properties passed by the Forms Designer host application. This will default to 255 if the host application does not set these properties. The TestGUI application obtains the sizes from the FormItem table’s definition.

## Chapter Headings Record

The chapter headings record is used to provide a string that contains headings for each chapter in this encounter. It is a tab-delimited string where each sub-string is a heading of a chapter; the position of the sub-string identifies the number of the chapter it is a heading for.

## Sub-Group Headings Record

The sub-group headings record headings record consists of the tab-delimited string of sub-group headings. SubGroup captions are contained in a tab-delimited string with each caption having the syntax: 'sgn-caption'. 'sgn' is the value of the subgroup number for which the 'caption' applies. The sequence of the substrings within the string determines the presentation order of subgroups to which they apply. Note that these captions are applicable only to findings where the application has assigned one of the miscellaneous narrative groups

## Chart Finding Record

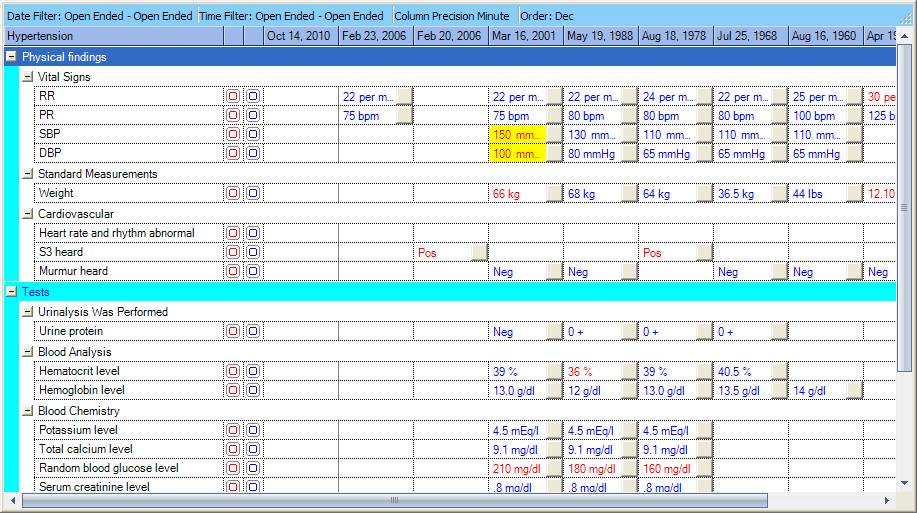
The chart finding record consists of the following comma-delimited fields:

1. UniqueID: is the unique identifier required for data entry.
2. Seq: contains the user’s assigned sequence for a finding in the assessment narrative group.
3. Medcinid: is the permanent Medcin identification number.
4. Flag: stores a bit structure used by the GUI modules to further classify the finding. Bit assignments are as follows:

* None = 0: no flag bit set.
* ROS = 1: Show finding as ROS, not HPI.
* ChiefComplaint = 2: Show finding as a Chief Complaint.
* TextFirst = 4: Show free text finding to be narrated at beginning of group.
* Active = 8: Show finding as active problem.
* Chronic = 16: Show finding as chronic.
* NewProblem = 32: Show finding as new to provider
* WorkupPlanned = 64: Show finding as additional workup planned.
* Form = 128: Show finding as entered from a pop-up form.
* Changed = 256: Used internally; finding has been changed..

1. Prefix: is the Medcin prefix qualifier code string.
2. Modifier: is the Medcin modifier qualifier code string.
3. Result: is the Medcin result qualifier code string.
4. Status: is the Medcin status qualifier code string.
5. Episode: is a string that contains a packed version of the episode fields; the format is “expression|FreqFm|FreqTo|DurFm|DurTo”. Expression is “F” for *frequency* or “I” for *interval*. Each additional field is formatted as a series of digits followed by a time unit character (“N” minute, “H” hour, “D” day, “W” week, “M” month, “Y” year). The string is converted to presentation format for narration.
6. Onset: is the onset of the observation in Medcin date format, (YYYYMMDDHHNN).
7. Duration: is the duration of the finding maintained as a unit of time (a number followed by a unit of time specification character).
8. QualifierLink: is the code string assigned for linking two findings, the link group code assigned when grouping two or more related findings, or a CCC code indicating a linkage to an RNDx finding.
9. QualifierLinkSequence: is the UniqueID number of the linked finding.
10. Referid: is the id number assigned to a physician who is referred the patient for this finding.
11. Value: is the measurement value assigned.
12. Unit: is the unit for the value field.
13. RangeNormalLow: is the normal low value of the range.
14. RangeNormalHigh: is the normal high value of the range.
15. RangeScale: is the calculated range scale.
16. eDocId: used to set the DoctorID when the doctor entering the data is not the DoctorID as specified in the Encounter record for the encounter.
17. ReferenceID: is the reference identification for tests and therapies.
18. Note: contains the free-text note attached to the finding.
19. Group: stores the Narrative group if assigned by the application or pre-assigned in a form, otherwise the value stored is 0.
20. Chapter: is available for storing an encounter chapter assignment.
21. UserSubGroup: is a property an application assigns to a chart object provided the Group property is one of the miscellaneous narrative groups.
22. EditDateTime: This contains the last date and time that a property of this object was changed. It is up to the application to maintain this property
23. Codes: This property contains codelink codes assigned during user session. The syntax of a Codes string is as follows: “codetype,code[;codetype,code]”
24. Specifier: This is a semicolon delimited list of codespec lists. A codespec list is a comma-delimited list of codes. The specifier property can contain more than one list of codes, with each list delimited using a semicolon. The specifier starts with a field indicating the codetype of the code used to obtain the lists (ex: 23;A,B,D;AB,C,DA). The specifier is used in conjunction with a DSM-IV code to add additional uncoded textual content to commonly used in a finding's narration. The lists would be in sequence with a corresponding codespec record for the DSM-IV code in question.
25. Timing: This is the HL7 timing component assigned for special orders; it is currently used in the nursing module for the assignment of a SIG to RNOx nurse orders.

# Flow Sheet View



*Example of a Flow Sheet from the Forms Designer*

# Miscellaneous Notes

Copied and pasted from other documents, for my reference as I integrate them into the note above.

## Calculated findings (e.g. BMI), and MedcinID codes for weight, height, and BMI values

* Active Forms has a feature where a single finding can be marked to automatically have its value property calculated at run time from the other findings enclosed in the same frame. The calculation can be either the sum of all the other findings, or the Body Mass Index (BMI) calculated from the weight and height findings enclosed in the same frame. You can designate that calculation operation finding by checking the box labeled “Calculation Value in Frame”. Note that all the source findings to the calculation as well as the marked finding must be in the same frame, and all findings must incorporate numerical-only values. The calculated value is changed only during data entry, so if a previous encounter that includes the set of findings related to the calculation is loaded, the calculated value must be correct in the set. In the case of BMI, you must have only one weight finding (Medcinid 6063) and one height finding (Medcinid 6071), as well as the BMI finding (Medcinid 9300) enclosed in the frame and no other findings. At runtime, the Active Forms module will establish the correct calculation to use based on the contained findings.
* Mathematical equations example can be found in the DotNetUsersGuide on page 103

# Things to Investigate

## Unsorted

\* ItemData/unique identifier (N=<int>): controls are assigned sequentially, but their sequence in the exported template file does not affect layout. Is this used meaningfully anywhere? (YES – answer was in the dotnet guide, but forgot where. It affects equations, which include the N number as part of the identifier)

\* TabStrip/ItemData/(HHL=) first showed up in the FWST forms

\* TSWFPGEN started showing up in Prefix[4]/Value - what's this do?

\* z-ordering: ribbons cover anything that doesn’t list them as a parent ribbon. Order in the template file does not affect z-order.

\* Lastly, is it possible to have radiobuttons linked to a multiline textbox, with each radiobutton tied to the same MedcinID, such that switching radiobuttons populates the textbox with their predefined content? (Or each has its own textbox that overlaps with the others, and is raised to the top of the z-order when selected). This would be very useful for multiple versions of a provider's default text, say, a free-texted "normal" physical exam for infant, child, adult.

\* weird textbox behavior (arrow key events being disrupted etc)

\* any transforms happening within controls en route to note output

## Difference between "Include in AutoNeg" and not included (DotNetUsersGuide, p81)

"The Active Forms module supports the ‘autoneg’ functionality for a block of symptom and/or physical exam findings. The algorithm is used to automatically set negative the top hierarchical finding of any node chain provided no finding in the chain has an existing data entry. The “Include in Autoneg” checkbox allows the designer of a form to force the runtime module to include the given finding autoneg consideration even if it is not the top hierarchical finding of a node chain. Invoke this property by checking the box; it will be enabled only for symptom and physical exam findings."

... does not appear to affect whether something is included in copy-forward settings

## Prefix | Modifier | Status codes

• You can change the prefix by selecting an entry from the drop-down “Prefix” list. If the prefix is changed to other than blank, the user will not be allowed to change the prefix at runtime.

• You can change the modifier by selecting an entry from the drop-down “Modifier” list. Note that when a modifier other than blank is selected and the finding is entered in the forms runtime, the finding will be entered with that modifier. However, the modifier is not used as a property for matching when populating a form with an existing encounter.

• You can change the status by selecting an entry from the drop-down “Status” list. Note that when a status other than blank is selected and the finding is entered in the forms runtime, the finding will be entered with that status. However, the status is not used as a property for matching when populating a form with an existing encounter.

## Setting up a multi-column page with expanding ribbons contained within the columns

## Rich Text Box

The use of a rich text box control as an alternative to a text box control associated with a Medcin finding object on the form may be enabled by the application. Text boxes on forms are used to enter text that appears as a free text addition to a Medcin finding. The Medcin server’s narration engine does not provide any support for handling rich free text other than by accepting a segmented note string that includes marked segment for each narration output that an application wishes to utilize. The server’s narration engine has output options of plain, RTF, and HTML. With this in mind, the use of a rich text box control in a form design has been implemented with specific limitations. Specifically, it supports the use of a single font with any mixture of plain, bold, italic, underlined, and strikethrough font attributes. No other RTF commands are supported. The font used stems from the current user preference RtfStyle for the free text element. Furthermore, since a rich text box control uses considerably more computer resources than a regular text box control, the form design is limited to no more than 10 rich text box controls per form design. Also, you cannot pop up a stand-alone note dialog by double clicking the rich text box component.

When entering text into a rich text box control, font attribute selections are performed using a combination of the CTRL key along with one other key: B=bold, I=italic, U=underlined, and S=strikethrough. Styling can be applied either by selecting a section of text then typing the appropriate key combination, or simply by entering the key combination when no text is selected to change to or from the selected attribute. A bulleted list can be initiated by typing the key combination CTRL-L when no text is selected. If text is selected when CTRL-L is typed, the selected text’s styling will be toggled to/from bulleted. The use of the TAB key will move the cursor away from the rich text box and complete your data entry. You can advance to the next line by using the ENTER key; in contrast to a text box, the ENTER key in a rich text box control will not advance the cursor to the next control in the TAB sequence.

The chart object for the finding will retain the note text in a combined marked string containing plain text, the RTF, and an HTML equivalent of the RTF.

Would you be willing to make quick demo templates of a few features? There are a couple I can't find in the wild.

\* a Pick List with the Drop-Down List style - curious to see how it links to text

\* a Pick List with the Internal List style - to constrain the user to a few medcin findings

\* a picture - curious how the template links to the file location, or if it uses an internal list of pictures, how to access those

\* the logic display indicator – where there is a frame containing 5 findings, and the indicator/LED icon activates when >= 3 findings have been clicked

\* a multicolumn page with ribbons spanning across different numbers of columns

\* a simple flowsheet

\* a rich text box as an alternative to a textbox control