

This paper was written to document a system I devised for a Midwestern university. The documentation fields referred to all relate to various scores, grades and other types of documentation required as part of a prospective student's application for admission to the university.

THE NOTIFICATION SYSTEM: The University's GoldMine for Prospects contains about two dozen "documentation" fields, and one Status field. These fields are spread over a number of different Views, and typically may be edited during the daytime by people who receive actual documentation correspondence. Also, some information (for example, the waiver of a specific piece of documentation) may be entered by others.

This updated information needs to be called to the attention of Counselors who work in the evening, phoning Prospects who have just submitted some documentation information. This seems to be the best time to get action on documentation requirements that are still unmet. Each Prospect is assigned to a specific Counselor, and the GoldMine Username of that Counselor is stored in a field of the Prospect's record.

To facilitate the Counselors' work, a Notification System is needed. The objectives of this system are two-fold:

1. **Create a single display** that contains existing information extracted from all the documentation fields (and the Status field); and which includes the **date on which the current value in the field was acquired**. There are about two dozen of these documentation fields. There are currently **no date fields** associated with the **documentation/Status** (henceforth called **D/S**) fields to capture the date edited. Part of the requirement for this display is that, in the event that the value in a documentation field changes from its first-entered value, **all** values (previous and current), **along with their date of entry**, should be shown in this display.
2. **The second requirement** is that, for any Prospect whose record has been edited in any of the D/S fields on a given day, that Prospect is to be included in a special new "Notification" Group for the appropriate Counselor, for that day.

So, in summary, we need a system that displays all the D/S data that exists for each Prospect, along with its date of entry; and which will direct each Counselor's attention to the just-edited Prospects belonging to that Counselor.

WHAT WAS CONSIDERED AND REJECTED

- GoldMine offers the option to create a **History Log** every time a field is changed. This was rejected on two counts. One, entering the initial value in a field does **NOT** create a History Log record, and that was not acceptable in this situation. Also, using the History tab for this purpose would not effectively isolate the desired information, and that was a practical requirement.

- Adding **Date Type fields** for each of the D/S fields was considered. But Admissions didn't want to add that many more fields to the database. Also, this would have required creating a GoldMine View with two dozen D/S fields, plus an equal number of Edit Date fields for them; and the Counselor would still have had to scan through all those Edit Date fields, just to find the documentation fields that were most recently edited. Finally, the requirement for tracking changes in values would not be handled.

WHAT WORKS

This is a classic application for GoldMine's **Expanded Detail records, using a dedicated Custom Tab**. Here's why:

- Provided all the Details are the same Type (we used "Action for Counselor" as the Type), they will all fit into a single Custom Tab display. That meets the **single display** requirement in item 1, above. We named that Custom Tab "Actions". We used the **Reference field** of the Detail to store the **specific D/S Type**; and an expanded field (the Title field, relabeled as "Value as of Date") to store the **actual value for that D/S type**.
- All Details have a date associated with them, the date on which they were created. Also, within a Custom Tab, **Details are sorted by that date**. This means that the arrangement of the Details under the Tab will automatically follow a natural chronological order, as requested by Admissions¹.

Of course, all this must be automated in order for it to be practical. Admissions wanted a single button to click that would make all the Actions Tab entries required each day; and to create the Notification Groups for all the Counselors for the day.

This has been accomplished using a utilities program for GoldMine, called GoldBox. When that requested button is pushed, GoldBox will:

1. Create a list of the Prospects for whom any D/S field was edited since the prior run of the system. GoldMine's Lookup.ini file tells us which Prospects have had a D/S field edited each day. The actual values for ALL the D/S fields for each of these Prospects are included as part of the list, whether just edited or not.
2. For each of those Prospects, compare the data collected from the D/S fields to the data under the Actions Tab. New Tab Details will be made whenever either:
 - a. A specific D/S Type is found in the list, but not under the Tab. Or,
 - b. A specific D/S Type was found under the Tab, but with a different value than the value in the new list; and no Tab entry could be found with the current D/S value.

Every item in each day's list is evaluated against the appropriate Prospect's Tab every day. So, when any of a Prospect's D/S fields are edited, the Counselor will have the very latest data available when viewing the Prospect's Action Tab.

3. After some additional processing, the Groups are created for each Counselor. The name of each Counselor's daily Group includes the date it was created; and each Group's Membership consists of the Prospects who had any D/S field edited since the previous Group was created.

¹ To put the system into operation, it was decided that, for any given Prospect, the Actions Tab would not be populated until one of the D/C fields has been edited, **after** the date on which operation of the system was begun. This means that Prospects having D/S data as of that date will have the Details for that pre-existing data added **only when a "new edit" to one of the D/S fields is made**. This is considered acceptable because the dates on which the pre-existing edits were actually made are not known, anyway.

Here is an example of an Actions Tab, with the Details beneath it. The data is purely test data, and is entirely meaningless.

The screenshot shows a software application window titled "Bob Taylor". The window contains a header with the student's name and contact information. Below the header is a navigation bar with tabs for "Details", "Pending", "History", "Links", "Members", "Tracks", "Oppty", "Projects", and "Actions". The "Actions" tab is currently selected and displays a table with the following data:

Date	Action for Counselor	Value as of Date
12/29/2008	HS Trans	W
12/29/2008	Stage	28
12/29/2008	Last Coll	F
12/29/2008	C3 Trans	F
1/1/2009	C2 Trans	F
1/1/2009	Evaluation	F
1/1/2009	C4 Trans	P
1/1/2009	First Rec	Y
1/1/2009	2nd Rec	Y
1/1/2009	Essay	W
1/1/2009	SAT Rept	Y
1/1/2009	ACT Rept	Y
1/1/2009	Housing Form	N
1/1/2009	Housing Pref	2
1/1/2009	Deposit Date	12/29/2008
1/1/2009	HS GPA	3
1/1/2009	App Fee	W
1/1/2009	Rank %ile	92
1/1/2009	ACT COM1	20
1/1/2009	ACT COM2	25
1/1/2009	ACT COM3	44
1/1/2009	SAT SCORE 1	1
1/1/2009	SAT SCORE 2	2
1/1/2009	SAT SCORE 3	32
1/1/2009	Visit Date	12/29/2008
1/1/2009	C3 Trans	P