

# Proposal to Revamp the CU\*BASE Menu System

## Designing Tools for the Google Generation

Presented by Dawn Moore, VP Writing Team/Product Design

Rev: January 29, 2015

This document outlines the basic goals for a complete rewrite of the CU\*BASE menu system, including new techniques for configuring the employee security settings that control what functions a credit union employee or other CU\*BASE user can access.

Specifications for the file and programming changes are being developed separately – this proposal is simply intended to document the ideas we’ve gathered over the years and the thought processes that went into the design.

Why do this project now? Because CU\*BASE should help people find what they need, quickly and intuitively. It should show them what they can do without overwhelming them with what they can’t...while still nudging them to try new things they *should* be doing. It should help CUs easily control what their employees can and cannot do, with less room for human error.

And it should show off our software more effectively to a new generation of tech-savvy employees.

<b>Project Goals</b> .....	<b>1</b>
<b>The User Interface</b> .....	<b>2</b>
Prototype .....	2
Goals for the UI .....	3
<b>General Notes</b> .....	<b>5</b>
Attacking the Project: A Team Effort .....	5
Changing Our Lingo .....	5
<b>Employee Security for CU Security Officers</b> .....	<b>6</b>
Ideas for the CU Security UI .....	6
Other Ideas to Consider .....	8
<b>What to show when a user logs in?</b> .....	<b>8</b>
Coordinating All of the Controls.....	9
How Operator Edition Fits.....	10
<b>Project Plan</b> .....	<b>12</b>

## Project Goals

(in no particular order)

- Serve the Google generation – searching is the new norm
- Adopt an “it’s me” focus with more user-defined preferences – “show me only what I need, the way I like to see it”
- Scale back the number of features a user has to wade through to find what they want

**FUN FACT:** As of 11/24/2014, there were **956** menu options (not including OPER or custom) and **51** speed sequences on CU\*BASE menus.

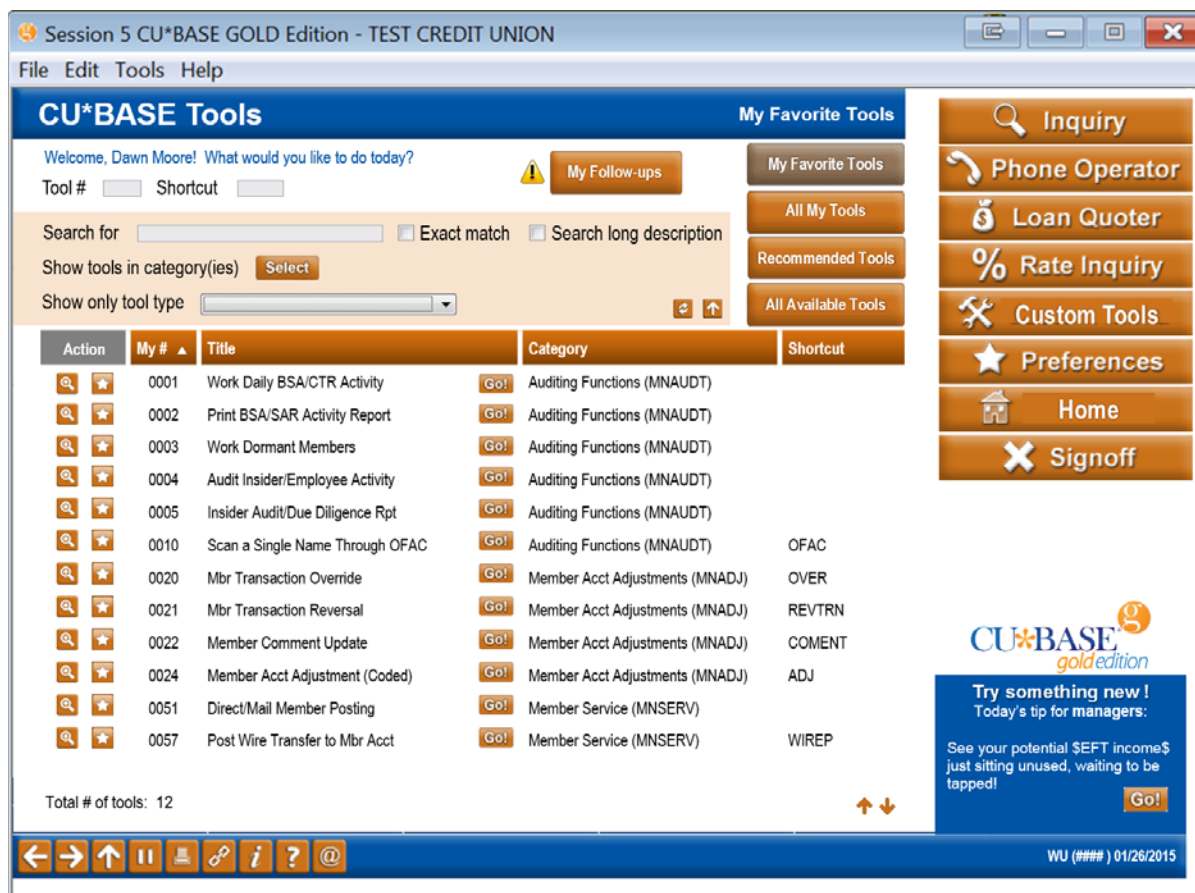
- Shorten the new user learning curve – “MNXXXX” is not friendly to new users; menu-driven system may be perceived as outdated
- Come up with a more intuitive way to separate normal features from the “dangerous” ones CUs shouldn’t be using
- To tie the security controls to the menu display, so the person can see just the commands they can actually use
- Reduce or eliminate the pain of menu changes for CU Security Officers (and free us from the restrictions imposed on us by our sensitivity to that pain)

- We've hit the limits of our infrastructure; new menus get added just because we ran out of room on a menu
- Operator Run Sheets and other instructional doc – why should we have to keep rewriting these over and over just because something got moved around?
- Better use of available screen space
- Allow for fast, user-customized keyboard access but also move towards more mouse-driven functionality that is faster and more intuitive
- Create a new style of customization so keyboarders can assign their own numbering scheme that will never change unless they want it to
- Facilitate our long-term vision for the **CU\*BASE Operator** edition look and feel
- Facilitate our long-term vision for the **CU\*BASE SE** edition look and feel

## The User Interface


### Prototype

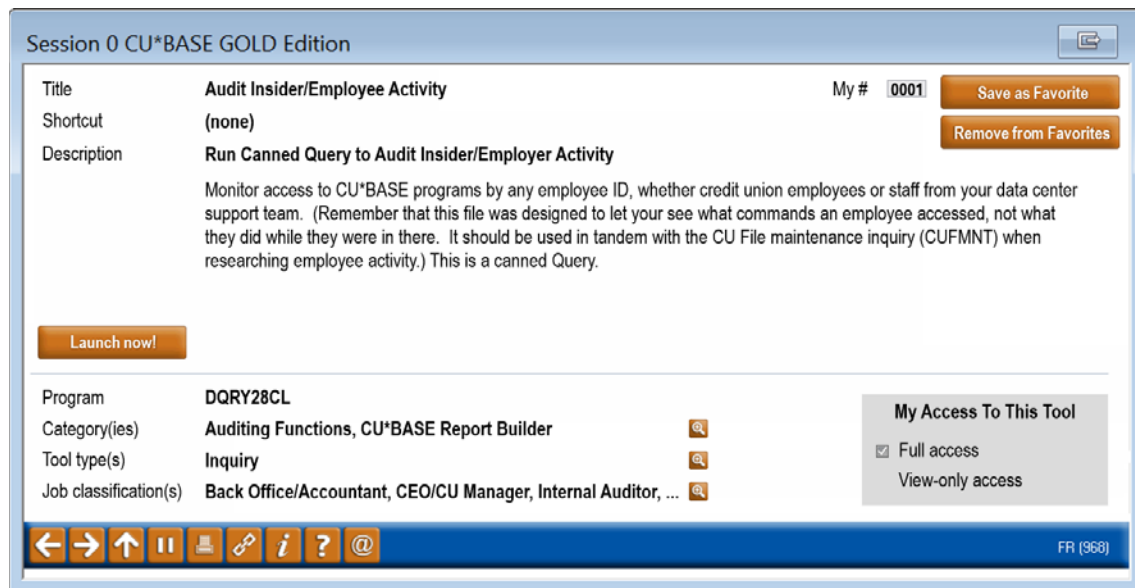
Below is a mockup of what users would see when they first log in to CU\*BASE:



There will no longer be other menus – this is command central, and from here a user can locate, read about, and navigate to any function CU\*BASE has to offer. For a list of other ideas that were considered when developing this prototype, see the next page.

## About This Tool and My Favorite Tools

This window would appear when the user clicks the Favorites button  on any item:



This would be a new host screen to display details about the particular program. It also allows the employee to add this command to their favorites and re-sequence the assigned tool# used for navigation.

## Goals for the UI

These are ideas that were compiled and used when developing the above prototypes. Not every idea necessarily made it into the initial design, and there are some ideas for future enhancements to revisit later.

- Need a visual indication of commands to which the user has not been granted security access. So the menu list would chain out to SECMAS automatically and allow the user to:
  - Show only options the user has flagged (“My Favorite Tools”)
    - OR
  - See only options the user can actually access (“All My Tools”)
    - OR
  - Show all options we’ve tagged for their particular job category (“Recommended Tools”)
    - OR
  - Show all options but gray-shade the ones the user can’t access (“All Available Tools”)
- Sort capabilities
- Shortcut/speed keys still (eliminate the extra security, though) – expand to allow a shortcut to any command that is commonly used (like items that are duplicated on many different menus now because so many different job functions use them)
- Some sort of job description-related presentation so the first screen you see when you log on is related to your job in addition to your specific authorities – *teller starting point, back office employee starting point, operator edition starting point, DC support staff/network community controls*
- Needs a really good, really easy search tool - Like Google, searchable by just entering a word or phrase, could limit search with filters but make that optional and allow the user’s preferences to be saved for next

time. Should do research into other search tools that let you fine-tune your searches somehow. Our local public library catalog, for example has some techniques that might be useful:

Search for  in  Show  [Save my search preferences](#)

→ Then Favorites could allow you to save your preferences for the search method too (the defaults for these drop-downs)

- Search results should allow you to go to/launch but also allow for “show me more about this” to see the description as well as the keywords, user types, etc., from SECMAS<sup>T</sup> settings
- Favorites/Preferences
  - In addition to the automatic filters according to SECMAS<sup>T</sup> settings (i.e., can I actually access this command or not), would still want a way for the user to specify their favorites, and choose the numbering sequence for those items so they can memorize the navigation like they can now if they want to
  - Perhaps a quick button/drop-down list to choose between “Show My Favorites Only” and “Show All available applications” and “Show all OPER applications” or something like that
  - Would also still be used for color theme, auto-security settings, etc. like now
  - Way to choose the default search criteria (i.e., titles only vs. descriptions, etc.)
  - Default to my favorites vs. all my tools
  - Favorite default sort order?
- Special presentation for new users or certain job descriptions (managers, for example) where we want to ensure they know about all of the available tools (see also “Blue-sky Dreaming” below)
  - For users with no preferences set up yet, would default to all my tools
  - Need to think through the user experience for the first time logging in, what takes the place of the MNMAS<sup>T</sup> “Main Menu” starting point? How does a user know what tools they should be using?
- Asked Mike about the possibility of an auto-complete/position to feature like Windows has, so as you typed the first couple letters of something into a search field the list would automatically jump down to that area. Apparently GOLD can’t do that so we’d have to include a host position to field (with Enter) if we wanted this.
- Way to “add to favorites” and “remove from favorites” from the sub-file list
- Advanced search ?
- Stats about # of items in each list, # of potential items, etc.
- Preferences ideas - would like to control by user ID (at the host level, so need a new file):
  - alternate-line shading on/off (??is this even possible??)
  - Way to choose a favorite search criteria (i.e., titles only vs. descriptions, etc.)
  - Default to my favorites vs. all my tools
  - Favorite default sort order?
- “Recommended tools” for my job description – make better use of the existing but weak job category (perhaps “job classification” to differentiate from the new program categories) – may need to keep/expand the old CU-defined one *and* add a new one for our purposes, tho’

- Will still need three separate host menu screens to facilitate the different look and feel for GOLD, OE, and SE (proj#37369)
- Randy's list:
  - New tools to display "what else is available to this CU employee"
  - New tools for "learn this category of software options"
  - New tools that compare CU Employee Templates by Position to the employees actual configurations.
  - New tools that show actual usage of the features during the month, frequencies, etc.
- Hover help or context-sensitive help clicks (to explain the categories, tool types, etc.)
- What about leaving room for CUs to leave daily messages to their staff?
- Buttons for "You have follow-ups" and other employee alerts?

## General Notes

### Attacking the Project: A Team Effort

Although specs are well underway, it's worth mentioning that this project, more than most, will require some creative collaboration with skilled program analysts. Their expertise will be important in defining the database structure that can support the vision of the prototype. This project will likely require some creative new techniques for searching, setting up favorites, and other functionality that to date has either been done by GOLD .NET programming, or has never been done at all.

- Dawn Moore – overall design and end-user requirements
- Deb Stickler – database infrastructure
- Bob Colburn and Jack Carpenter – software analysis
- Jeff Miller (and Jack Carpenter) – representing the Ops point of view
- Mike Warren and Amanda Grieves – user interface infrastructure

### Changing Our Lingo

From Menus → to My CU\*BASE Tools

Haven't come up with the definitive name yet, but the idea is to eliminate the concept of menus altogether. Instead, after a person logs in to CU\*BASE they will be presented with their **home page**, which is their access point to *everything* they can do in CU\*BASE.

Users running Operator Edition would see a completely different home page, with access only to Ops-related functions, but could toggle to the standard home page as needed.

<b>DC</b>	Data Center
<b>PC</b>	Production Center (Use "Production Center" in place of "Ops" too?)
<b>OE</b>	Operator Edition
<b>SE</b>	Servicer Edition (CU*BASE for CUSOs)
<b>CU</b>	Credit Union
<b>Emp</b>	Employee

Use "tools" or "features" or "applications" instead of "commands" ?? (rejected "options" as too ambiguous) "Tools" could be confused with our "Toolbox" tools – although even that label could use some help

Use “menu” only to describe the single menu screen that displays all of the user’s available commands – or change to something other than menu? I’m leaning towards home page, actually, but “page” is less accurate for CU\*BASE so perhaps just “home”?

## Employee Security for CU Security Officers

In order to support the new layout, we would need to develop a new MENUCTL table(s) to record, for every executable program name, the title, program call and parameters, key search terms and groupings (accounting functions, auditing functions, member service functions, custom programs, etc.), job categories, and other information for display to the user and/or use in organizing the feature on the home page. The security officer simply assigns access to a tool (program) and then, if applicable, chooses view-only or full access for each individual employee.

*And here’s my favorite part: we can now create a shortcut (speed sequence) for any feature we like, with no extra effort required on the part of the security officer!*

Since changes to program names are extremely rare, this would significantly reduce the amount of effort required on the part of a CU security officer when new features are introduced. It also eliminates the confusion that happens now when a person who can access a feature from one menu tries to choose the same feature on another menu (or enter its shortcut/speed sequence) and can’t, simply because of the way our security requires the CU to set up each occurrence separately.

But more than that, this new structure opens the door to many new techniques for how CU\*BASE is presented to end-users.

→ See the project specifications for a list of proposed fields for this new file.

## Ideas for the CU Security UI

- Much of the existing MNMGMT #2 program will remain pretty much as is, since the basic employee profile, special security, and account security isn’t changing. I think only SECMENU is changing, in fact. But we will use this opportunity make the maintenance program more of a dashboard with filters, better techniques for finding an employee in the list, etc.
  - TBD: I may also try to use this project to envision a technique for storing start and end dates for employee IDs, and perhaps a new way to pull “suspended” IDs out into a separate file for audit and historical purposes, freeing up the ID for a future new employee. (See the specs for more details.)
  - Also TBD: Lender\*VP also needs to add a field at the employee ID level for NMLSR # (for loan officers). Plus a NMLSR # at the CU level, perhaps in GLMAST. May include here or make a separate project, not sure yet.
- Include a way to select a program name and have the dashboard show you all of the users who already have access to that. At the very least will include that as a sort option on the security reports.
- When creating a new user, it would be nice to have a better wizard-type flow that makes it easier to set up the user based on job category and/or copying from another employee.
- Assuming we can figure out how to ascertain that the person logging in is the CU security officer, the interface should be able to show only tools that apply to the particular user types (i.e., if CU security officer, then only CU employee items would appear; if data center security officer, then DC employee items, etc.).

- Way to show “new” or “updated” program names in the list, for after a release? (For example, if the program record in the MENUCTL file had a “new” flag then this program could be smart enough to look for records with a date in that field and highlight them or something – need to think this through some more.)
- Allow “assign to employees” on each program that can (a) flood to all employees with this job classification or (b) multi-select employee IDs or (c) flood to anyone who has access to existing program xxxxxx (this last one is also cool because if we change a program name, maybe we could use this same tool to move everyone over to the new program based on the old program name, before deleting it?). Similar feature when adding a new employee ID.
- All 3 of the Employee Security reports on MNAUDT #26 would have to be rewritten to the new layout, such as the Menu Security report and the Employee Security report. Would be nice to have some auditing reports that group features somehow, so an auditor could see, for example, all of the programs considered “configuration” and who has been authorized to access them.

### Organizing Access to Programs by User Type

Per Jack, OPER menu security can only be set by someone with QSECOFR authority – so a CU employee can’t give themselves menu security access to MNOPxx menus. I’m hoping this proposal could make that restriction unnecessary.

One thought was to include a flag in the new MENUCTL file for a “user type” (credit union employee, Data Center employee, and/or Production Center employee). Then when the commands are listed in the employee security config, the display takes into account the user and only shows the appropriate ones.

*Don’t know if we also need another level of operator: Supervisor or something?*

*Might also need to think these through specifically for self processors – are the terms useful for them, too?  
See Page 8 for a discussion of what we’d need to do upon login in order to support this idea.*

For example, MNCNFB stuff would show:      Whereas MNOP09 stuff like CU Master Parameters would show:

- |  |  |
|--|--|
| <input checked="" type="checkbox"/> CU employee          | <input type="checkbox"/> CU employee                           |
| <input checked="" type="checkbox"/> Data Center employee | <input checked="" type="checkbox"/> Data Center employee       |
| <input type="checkbox"/> Production Center employee      | <input checked="" type="checkbox"/> Production Center employee |
| <input type="checkbox"/> Programmer/Admin (?)            | <input checked="" type="checkbox"/> Programmer/Admin (?)       |

This would control the SECMAST-type controls a CU security administrator uses – they would not see the DC emp or PC emp ones, same as now.

I would still like a clever term instead of “OPER” to describe the system/ hidden ones that only DC employees and self processors can get to. Also need some way of dealing with the features CU\*Answers programmers (or perhaps Site-4) might use but we don’t want self-processors to get to. That was my thought behind a possible “programmer/admin” type. Still needs thought.

When showing the CU\*BASE home page, the default display is to show only the commands that person can actually use. But when they click to show “all available tools” they would see anything else they might be eligible for – according to this user type control – but don’t happen to actually have access to. **That way it would not be necessary to actually GO to “OPER” to see config features any longer!**

## Other Ideas to Consider

- What about “other things you should be doing” suggestions based on the new categorizations in MENUCTL? For example, when a manager signs on, any dashboards that aren’t currently in their favorites or for which they do not yet have security access, would appear in a sort of “have you tried this dashboard yet?” ad space →
- Allow a space on the menu for CU-defined messages to their staff?
- A place for us to advertise upcoming releases, maybe alternating with a “Remember this...” or “feature of the day” spot?
- “You have follow-ups” - Tracker stats with click direct to program?
- Would the new infrastructure make it possible to change how we do the “Special Security” settings (for features that don’t use ESEC) ?? Can’t really imagine it yet but perhaps we can ultimately streamline and simplify how employee security is set up by allowing those to be listed along with “menu options” when assigning access or on auditing reports.
- Some sort of “flood” programming in MENUCTL maintenance program that would allow me to move all security from the old program name to the new one automatically when menu changes were implemented? (Might not happen often enough to matter, could wait for future enhancement.)
- Incorporate add’l controls for OPER and SYSCTL-type menus too, maybe? & toolbox? & SUPP menus on DEV and QC?
- Add flag to Network Communities config so by User ID you’d be able to say which menu system they get upon logon ?? Or even by User ID/CU name so you could be an Operator for one CU and normal CSR for other CUs? (thinking of Site-4)?

**Try something new!**  
Today’s tip for managers:

See your potential \$EFT income\$ just sitting unused, waiting to be tapped!

**Go!**

## What to show when a user logs in?

A key component of this design that’s still being determined is, when a user logs in, how do we know which home page to display, and what should be on it? Is it CU\*BASE GOLD or CU\*BASE SE or CU\*BASE Operator Edition? Which features should display when the user clicks “Recommended Tools?”

This will require us to **identify the type of employee logging in** and compare against which types of employees are authorized to see which groups of features.

Since Employee Security settings (SECMAS) are by employee ID, a lot of these decisions must be based on that, but there’s also the overall type of user logging in (DC employee vs. CU employee, for example). And the unique needs of self processors vs. Site-Four vs. CU\*Answers employees. And how to allow a person to play more than one role (like a DC employee who also works in the Call Center).

This section is just my brain dump of things to think about and somehow consider in the design.

For example, because we only know what programs they have access to based on the employee ID, then **auto security** becomes pretty important, because we can’t do anything with favorites until we know who the

employee is. But auto security is a no-no at some CUs. So my idea is to combine auto security with a new login procedure where we always ask for an employee ID and password upon login. Then, if the user engages auto-security, we have the ID we'll need for that, and it will work the same as now. But if they don't (or can't), we still have the ID we need to know which tools to show on the home page, and the system will require them to re-enter their ID and password every time they actually want to launch an application.

## Coordinating All of the Controls

**Network Communities, Data Center vs. Production Center vs. CU Employees, and more...**

Whether a person gets CU\*BASE GOLD, CU\*BASE SE, or Operator edition as their "home" menu should be controlled at either the workstation or user/employee ID level – need some help thinking thru the coordination of all these variables:

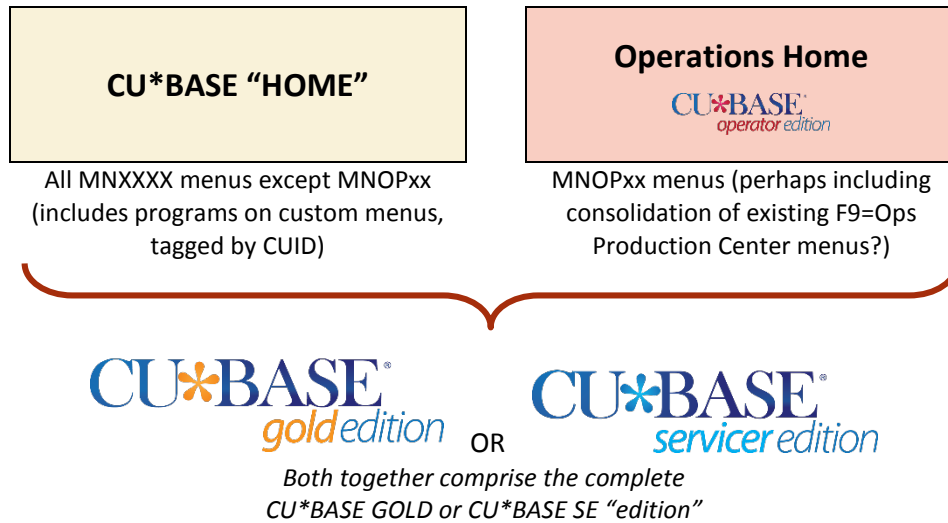
- Network Community controls
- System i user profile controls, including "initial program to load" and job description and other security controls
- CU\*BASE Employee ID vs System i user profile
- Data center employee aliases
- Existing SECMAST job descriptions
- CU\*BASE "edition" loaded on that workstation (SE in particular)
- User type (i.e., normal or operator or any combination – see below)
- Different behavior by CU – i.e., CMS or Xtend employees who play different roles at different CUs (see below)
- Self processor employees – operators versus their management (see below)

For the UI, will need a treatment that will make sense and be easy to use for all of these different types of end-users (and remember people who play more than one role, depending on what work they are doing):

- CU employees (tellers, MRs, accountants, etc.)
- CU managers (CEO, supervisors, people who can do configurations, etc.)
- Data Center support staff - level 1 (CSRs)
- Data Center support staff - level 2 (programmers, etc.)
- Data Center/Production Center Operators
- Site-4 support staff
- Site-4 operators
- CU Security administrators
- QSECOFR and other "special" user profiles
- CMS and Xtend employees working as a CU employee
- CMS and Xtend employees working as DC support staff
- Self-processor CU employees
- Self-processor Operators

## How Operator Edition Fits

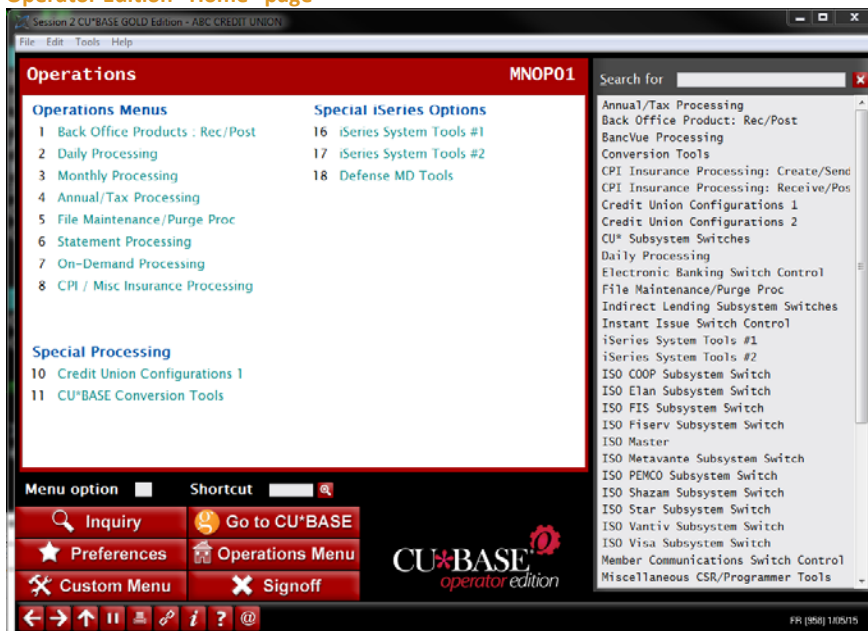
There will now be two main “home” pages (menus):



Note: Although it’s referred to as Operator “Edition,” OE is really always just a sub-set of the entire software, since even a user that starts out directly on the OE home page can always navigate back and forth to CU\*BASE (GOLD or SE, whichever edition their organization happens to be using).

As already mentioned, we’re still figuring out the best way to determine which starting point to use based on the individual user. We also need to address the easy navigation between both (or all three), **while still maintaining a completely different look and feel** for the GUI design for OE versus whichever version of CU\*BASE the user is using (GOLD vs. SE).

### Operator Edition “Home” page



This OE prototype uses the existing MENUCTL structure, with changes that allow a user to navigate to other menus via the MNXXXX shortcut. We could theoretically continue to use this old structure, at least for a while, navigating between the two separate menu systems. But even if we transition all of the commands currently on

MNOPxx menus (and perhaps the F9=Ops menus, even), we need to maintain a completely separate look and feel, even if that means just rearranging items on the panel and omitting functions (like Phone Op, Loan Quoter and Rate Inq) that are irrelevant to operators.

**Other tools to include in the future:**

Although we won't address any of these in this first iteration, while we are designing the database structure we are keeping these other "menus" in mind, with the idea that we might be able to use the same core database and controls to present many other groupings of tools that are currently handled separately.

- F9=Ops menus? Currently these are redundant to OPER menus, and there is variation from one data center to another, and it's all very confusing and impossible to document. Would this new infrastructure allow us to streamline any of this redundancy?
- SUPP menus? Would be nice to make it easier for a non-programmer to manipulate those menus as file libraries and special setups are added/removed
- Conversion tools? MNOP10
- CSR/programmer tools? MNOP21
- Other tools programmers use that aren't anywhere on menus yet because we don't want self processors to get to them (master config maintenance programs, data repair programs, etc.)
- Toolbox menus? - consider wrapping these in somehow?

## Project Plan

Following are the key milestones for the completion of this project:

Task	Target
<input checked="" type="checkbox"/> Present an overview of the design to Deb S. to get input on structural design <input checked="" type="checkbox"/> Project #37579 created (to allow Deb to book time for design work) <input checked="" type="checkbox"/> Present an overview and plan to Executive Management	By January 31, 2015
<input type="checkbox"/> Present an overview to Jack C., Jody K, and Bob C. for assistance in finishing specs and estimating development effort/timeline	During February
<input type="checkbox"/> Finish preliminary/working specs	By March 31, 2015
<input type="checkbox"/> Development begins	TBD (April?)
<input type="checkbox"/> Refine the prototype graphics for presentation to clients <input type="checkbox"/> Introduce project concepts at the Leadership Conference	By June 17, 2015
<input type="checkbox"/> Development continues, with periodic touch-base meetings between developer(s) and designers to work through the holes in the preliminary design and make decisions	TBD (June thru ??)
<input type="checkbox"/> Present updated designs/working prototypes during CEO Strategies	November 10-12, 2015
<input type="checkbox"/> QC Testing	TBD (Nov-Jan?)
<input type="checkbox"/> Documentation / Rollout communications	TBD (Jan-Mar?)
<input type="checkbox"/> Implementation	<b>Spring release 2016</b>