



# Generic UDC Code

By Mike Savino

**Editor's Note:** The ability to create a new "custom" UDC Table in JD Edwards is cause for celebration! If you're creating custom programs, the custom UDC helps you avoid the necessity for hard-coded values in your programs, while increasing flexibility and giving the custom work a JD Edwards look and feel that your end-users have become accustomed to. This white paper includes a nice example of how this technique works, and includes all the code you need at the end!

## Overview

Since the JD Edwards-supplied user interface program P00051 can be used to maintain values in any UDC Table, you'll spend less time on simple file maintenance programs and more time developing solutions to business problems. Just about the only downside is that someone has to enter and maintain the data in the UDC.

Depending upon your security policy, this can mean more work for your support resources. If you put the maintenance in the end-users' hands, there are a limited number of entries to start with, and the data is fairly static, then chances are that they won't complain too much about the chore. There are times, though, when your UDC Table could have many, many values, thus making the data entry work tedious, time consuming, and potentially error prone.

So what, if anything, can be done to reduce the number of entries? Sometimes, the answer is "nothing" if there is no particular pattern to the entries in your Table. But if you can express and group the entries into ranges, then the utility program in this article could save you and your end-users a lot time and effort. The program makes it simple to do a Generic search, and with a little effort, can even be made to work if there are a few entries in the UDC that are exceptions to the generic pattern.

Postal codes and general ledger accounts are good candidates for using a generic search. In the example below, the UDC Table in Figure 1 matches a municipality to strictly fictitious postal codes and postal code ranges.

```

00051          General User Defined Codes
                System Code. . . . . 55
                User Defined Codes . . . . ZT
Action Code. . . . . I          Skip To Code . . .
                                postal to city test

  05 Character
    Code           Description           Description-2
  11***          NEW YORK
  11234          QUEENS
  22***          CHICAGO
  33***          SEATTLE
  33456          REDMOND
  44***          BOSTON
  445**          LYNNE
  44553          FRAMINGHAM

F5=Code Types  F14=Memo  F15=Where Used  F18=Translate  F21=Print  F24=More

```



## Generic UDC Code

Figure 1 – UDC Table 55 ZT

As you can see in Figure 1, the ability to use the asterisk as a generic character saves the client from having to enter hundreds of postal codes. Add in the fact that the most specific values are always tested before the most generic values means that Queens won't be lumped in with the rest of New York, and that Lynne and Framingham are kept separate from Boston. (Anyone who knows these cities knows this is a good thing!)

### Technical Points

The generic search utility program, X099UDC, uses the JD Edwards UDC server program X0005 and requires several JD Edwards-supplied copybooks in order to compile. So, in order to use the sample code supplied with this article, you'll have to be a JD Edwards World customer with source code installed.

X099UDC is called in much the same way as the JD Edwards UDC Server X0005. When the UDC entry is found, the entire UDC record is returned to the calling program. Several additional parameters are required for the generic search capability. An example of a program that uses X099UDC is included with this article and each parameter is documented within the program.

X099UDC can be used with any UDC Table. However, if the UDC is not set up for a generic search, your applications would be taking an unnecessary performance hit while the program searches for generic entries that don't exist.

When creating a UDC Table with a generic search capability, keep in mind that the generic character does not have to be the asterisk, but should be a character that would never otherwise be used in a valid entry. If this isn't self-evident when looking at the UDC, then document your generic character for the benefit of others who might be maintaining the UDC Table and for the developers using X099UDC.

The number of positions to check for the generic character should also be documented. If the program that calls X099UDC does not specify the number of positions to check, X099UDC will, by default, assume that a generic entry can consist of a real value in the leftmost position of the UDC Code followed by all generic characters. A generic entry of that nature may not be practical in many cases. By specifying the number of positions to check, program performance will be improved.

A generic entry should always be entered with a real value in the leftmost position of the UDC Code. An entry with all generic characters will never be found by X099UDC, even if the number of positions to check is set to the full length of the UDC Code. You see in Figure 1 how, at the minimum, the first digit of each postal code is entered, followed by the asterisk. Therefore, the UDC Code length must be at least 2 characters in order to use generic characters.

## This Article Continues...

**Subscribers**, log in from our main search page to access the full article:

[www.JDEtips.com/MyAccess.html](http://www.JDEtips.com/MyAccess.html)

**Not a Subscriber? Gain access to our full library of JDE topics:**

[www.JDEtips.com/JD-Edwards-Library](http://www.JDEtips.com/JD-Edwards-Library)

Visit [www.JDEtips.com](http://www.JDEtips.com) for information on the JDEtips University schedule, private training and consulting, and our Knowledge Express Document Library.

License Information: The use of JDE is granted to JDEtips, Inc. by permission from J.D. Edwards World Source Company. The information on this website and in our publications is the copyrighted work of JDEtips, Inc. and is owned by JDEtips, Inc.

NO WARRANTY: This documentation is delivered as is, and JDEtips, Inc. makes no warranty as to its accuracy or use. Any use of this documentation is at the risk of the user. Although we make every good faith effort to ensure accuracy, this document may include technical or other inaccuracies or typographical errors. JDEtips, Inc. reserves the right to make changes without prior notice.

Oracle and J.D. Edwards EnterpriseOne and World are trademarks or registered trademarks of Oracle Corporation. All other trademarks and product names are the property of their respective owners.

Copyright © by JDEtips, Inc.