

**CBVGET(3)**

| REVISION HISTORY |      |             |      |
|------------------|------|-------------|------|
| NUMBER           | DATE | DESCRIPTION | NAME |
|                  |      |             |      |

# Contents

|          |                     |          |
|----------|---------------------|----------|
| <b>1</b> | <b>SYNOPSIS</b>     | <b>1</b> |
| <b>2</b> | <b>DESCRIPTION</b>  | <b>2</b> |
| <b>3</b> | <b>RETURN VALUE</b> | <b>3</b> |
| <b>4</b> | <b>ERRORS</b>       | <b>4</b> |
| <b>5</b> | <b>EXAMPLE</b>      | <b>5</b> |
| <b>6</b> | <b>BUGS</b>         | <b>6</b> |
| <b>7</b> | <b>SEE ALSO</b>     | <b>7</b> |
| <b>8</b> | <b>COPYING</b>      | <b>8</b> |

## Chapter 1

# SYNOPSIS

```
#include <ubf.h>
```

```
int CBvget(char *cstruct, char *view, char *cname, BFLDOCC occ, char *buf, BFLDLEN *len, int usrtype, long flags);
```

Link with *-lubf -lnstd -lm -lpthread*

---

## Chapter 2

# DESCRIPTION

Function is used to get field data dynamically from VIEW buffer. *cstruct* is pointer to C structure instance (object), *view* is name of the VIEW (must match the instance of the object), *occ* is field array occurrence, if field is not an array or first array element must be read, then use *occ*=0. *buf* is buffer where to output the data. *len* is used on both directions, on **input** it is used for indicating the length of the bytes for the target buffer (*buf*). If the data is larger than *len* indicated, then error **BNOSPACE** is returned. Then *len* is optional. If set to NULL, then no such checks are made, and it is assumed that buffer length is sufficient to store the data. *len* is used only for **BFLD\_STRING** and **BFLD\_CARRAY** fields. On output *len* (if not NULL), indicates the length of data loaded in *buf*. Indicated in bytes. *usrtype* is user type indicating the data type of the *buf*, if types are different, the automatic data conversion is performed. *flags* is either set to 0 in which case data is returned as is in the VIEW, does not matter if it matches the **NULL** value specified in VIEW file or not, the exact value is returned. If flag **BVACCESS\_NOTNULL** is specified, then if value in field matches **NULL** column in view file, then error **BNOTPRES** is returned. **BNOTPRES** can be returned in case if *C* flag was set in view file, and the count indicator is set to less than occurrence specified.

---

## Chapter 3

# RETURN VALUE

On success, **CBvget()** return zero; on error, -1 is returned, with **Berror** set to indicate the error.

## Chapter 4

# ERRORS

Note that **Bstrerror()** returns generic error message plus custom message with debug info from last function call, stored in thread local storage (TLS).

**BEINVAL** *view*, *cstruct* or *buf* is NULL. *view* or *cname* is empty string (first byte 0x00). Or occurrence is out of bounds of the array declaration.

**BBADVIEW** invalid view name (not found in view file).

**BNOCNAME** field name (*cname*) not found in view structure.

**BNOSPACE** No space in *buf*.

**BMALLOC** Failed to allocate type conversion buffer.

**BNOTPRES** Field not present (in case field value matches NULL in view file and **BVACCESS\_NOTNULL** is set or count (C\_) field is less than occurrence requested).

## Chapter 5

# EXAMPLE

See `atmitest/test040_typedview/vaccget.c` for sample code.



## Chapter 6

# BUGS

Report bugs to [support@mavimax.com](mailto:support@mavimax.com)

# Chapter 7

# SEE ALSO

**CBvchg(3) Bvnext(3) Bvsetoccur(3)**

## **Chapter 8**

# **COPYING**

© Mavimax, Ltd