VSAM FILE TUNING, VSAM CATALOG DISPLAY, VTOC DISPLAY & MODIFICATION.

USERCAT CBLV11 (3380) TYPE NRECS PCNT ---- ALLOC TRACKS ---- FRSP LMAX

<table>
<thead>
<tr>
<th>Type</th>
<th>NRECS</th>
<th>PCNT</th>
<th>Allocation Tracks</th>
<th>FRSP</th>
<th>LMAX</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC.TEST.FILE.B</td>
<td>KSOS</td>
<td>189</td>
<td>59.1</td>
<td>10</td>
<td>5</td>
</tr>
</tbody>
</table>

**SEV 3-06** CA SPLITS TOO HIGH (1 PC OF INSERTS)
**SEV 3-08** CI SPLITS TOO HIGH (34 PC OF INSERTS)
**SEV 2-04** BUFSP TOO SMALL FOR EFFICIENCY
**SEV 2-11** INDEX CISIZE IS EXCESSIVE
**SEV 2-25** INEFFICIENT DATA CISIZE
* SEV 1-10 * IMBED COSTS 20 PC OF DATA SPACE
* SEV 1-18 * SEC EXTENTS EXIST

**WARN 016** LARGE ALLOC CHANGE

DATA

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CISZ</td>
<td>(12288)</td>
</tr>
<tr>
<td>CYLINDERS</td>
<td>(1,1)</td>
</tr>
<tr>
<td>FREESPACE</td>
<td>(32,10)</td>
</tr>
<tr>
<td>BUFFERSPACE</td>
<td>(26624)</td>
</tr>
</tbody>
</table>

INDEX

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CISZ</td>
<td>(512)</td>
</tr>
</tbody>
</table>

This document may be downloaded from http://www.cbl.com/documentation.php
## Contents

### Documentation Notes

- Summary of Changes ......................................................................................................................... 2
  - Introduced in Rel 9.8 (April 1997) .................................................................................................. 2
  - Introduced in Rel 2.00 (May 2001) ................................................................................................. 2
  - Introduced in Rel 2.10 (February 2004) ........................................................................................ 2
  - Introduced in Rel 2.12 (January 2007) .......................................................................................... 2

### Introduction

#### General........................................................................................................................................... 3
#### The CBLVCAT Approach .............................................................................................................. 3
#### The New User............................................................................................................................... 4
#### MVS Execution ............................................................................................................................. 5
#### Caution ......................................................................................................................................... 5
#### TSO Execution ............................................................................................................................. 5
#### VSE Execution .............................................................................................................................. 6
#### CMS Execution ............................................................................................................................ 7
#### Interactive Execution .................................................................................................................. 8

### Control Card Syntax Rules

- Introduced in Rel 2.12 (January 2007)............................................................................................ 2
- Introduced in Rel 2.10 (February 2004).......................................................................................... 2
- Introduced in Rel 2.00 (May 2001) ................................................................................................. 2
- Introduced in Rel 9.8 (April 1997) ................................................................................................. 2

### Guide to List Output

- Standard Reports............................................................................................................................ 11
  - Introduction ................................................................................................................................. 11
  - Example 1. Standard Catalog Report .......................................................................................... 12
    - Notes ........................................................................................................................................ 12
  - Example 2. Standard Catalog Report with OPTION .................................................................... 13
    - Notes ........................................................................................................................................ 13
  - Example 3. VSAM Volume Summary .......................................................................................... 14
    - Notes ........................................................................................................................................ 14
  - Example 4. VVDS Report .......................................................................................................... 14
    - Notes ........................................................................................................................................ 15
  - Example 5. ICF Volume Summary ............................................................................................. 15
    - Notes ........................................................................................................................................ 16
    - Notes ........................................................................................................................................ 16
  - Example 7. Standard Label Report ............................................................................................. 17
    - Notes ........................................................................................................................................ 17
  - Customised Reports .................................................................................................................... 18
    - Introduction .............................................................................................................................. 18
    - Example 8. Basic Customised Report ....................................................................................... 19
      - Notes ...................................................................................................................................... 19
    - Example 9. Selective Customised Report ............................................................................... 19
      - Notes ...................................................................................................................................... 20
    - Example 10. Advanced Customised Report ............................................................................ 20
      - Notes ...................................................................................................................................... 21
    - Example 11. Customised VVDS Report ................................................................................. 21
      - Notes ...................................................................................................................................... 22
    - Example 12. Combined VTOC Report .................................................................................... 22
      - Notes ...................................................................................................................................... 22
    - Example 13. Combined Catalog Report .................................................................................. 23
      - Notes ...................................................................................................................................... 23
    - Example 14. Combined Catalog and VTOC Report ................................................................. 23
      - Notes ...................................................................................................................................... 24
    - Example 15. Combined VTOC Report for all VTOCs ............................................................. 24
      - Notes ...................................................................................................................................... 24
    - Example 16. Combined Catalog Report for all Catalogs ....................................................... 24
      - Notes ...................................................................................................................................... 25
    - Example 17. Combined Report for all Catalogs and VTOCs ................................................ 25
      - Notes ...................................................................................................................................... 25
    - Example 18. Free Space across all VTOCs ............................................................................ 26
      - Notes ...................................................................................................................................... 26

### IDCAMS DEFINE and Reorganisation

- Introduction ....................................................................................................................................... 26
- Supported File Types ..................................................................................................................... 27
- Output File ....................................................................................................................................... 27
- Selection .......................................................................................................................................... 27
- Security Keywords ......................................................................................................................... 27
- Warning Messages ......................................................................................................................... 27
- IX Allocation .................................................................................................................................. 27
- Installation Standards .................................................................................................................... 27
- Example 19. DEFINE without TUNE ....................................................................................... 29
  - Notes .......................................................................................................................................... 29
- Example 20. DEFINE with TUNE ............................................................................................... 29
  - Notes .......................................................................................................................................... 30
- Example 21. Skeleton Reorganisation Jobstreams ..................................................................... 30

- Notes on Example 21: Skeleton Reorganisation Jobstreams .................................................... 30
# Contents

Guide to VSAM Tuning.................................................................33
  Tuning Considerations..........................................................33
    File Selection........................................................................33
    Environment Selection.........................................................33
    On-line and Batch files.......................................................33
    Alternate Index Files..........................................................34
    Database Files.......................................................................34
    Reusable Files......................................................................34
    Average Record Length Estimation........................................34
    Growth and Freespace.........................................................35

Tuning Output...........................................................................35
  Introduction.............................................................................35
    VERIFY Block......................................................................36
    TUNE Block.........................................................................37
    CAPacity Block....................................................................40
    JCL Override Block.............................................................41
    Example 22. Increased Initial Load.........................................42
      Notes................................................................................42
    Example 23. Limiting CL Size...............................................43
      Notes................................................................................43
    Example 24. Conditioning Distributed FREESPACEx...............44
      Notes................................................................................44
    Example 25. Conditioning Absolute FREESPACEx..................45
      Notes................................................................................45
  VSAM Monitoring....................................................................46
    Regular Monitoring.............................................................46
    Fine Tuning.........................................................................46
  VSAM Modelling.....................................................................47
    Introduction.........................................................................47
    Example 26. Modelling a File...............................................47
      Notes................................................................................48
    Example 27. Modelling for DASD Change............................48
      Notes................................................................................48

Summary of Syntax....................................................................50
  REPORT Summary.................................................................50
    Notes..................................................................................50
  OPTION Summary...............................................................51
    Notes..................................................................................51
  LISTVTOC Summary............................................................51
    Notes..................................................................................51
  LISTVTOC Summary.............................................................52
    Notes..................................................................................52
  LISTLABL Summary.............................................................52
    Notes..................................................................................52
  VTOC Modification Summary.................................................52
    Notes..................................................................................52
  Other Commands....................................................................53
    Notes..................................................................................53
  LISTVTOC Output Fields.......................................................53
    Standard Catalog Report....................................................53
      Notes................................................................................53
    Customised Catalog Report...............................................54
      Notes................................................................................54
  LISTVTOC Output Fields.......................................................54
    Standard VTOC Report.......................................................54
    Notes..................................................................................55
    Customised VTOC Report....................................................55
    Notes..................................................................................55
  Abbreviations and Synonyms................................................56
    Notes..................................................................................56

A-Z Reference...........................................................................57
  (Separator Character)............................................................57
  * (Comment)..........................................................................58
  ACCESSED (nn)......................................................................58
    Notes..................................................................................58
  ALIAS................................................................................58
  ALLFILES= YES....................................................................59
  ALLOC (nn)..........................................................................59
    Notes..................................................................................59
  ALLOCS (nn).......................................................................59
    Notes..................................................................................60
  ALLOCT (nn).......................................................................60
  Notes....................................................................................61
  ALLOCU (nn).......................................................................61

Notes......................................................................................52

Notes......................................................................................54

Notes......................................................................................56

Notes......................................................................................58

Notes......................................................................................59

Notes......................................................................................60

Notes......................................................................................60

Notes......................................................................................61

Notes......................................................................................61

Notes......................................................................................62
Contents

A-Z Reference

Notes .......................................................................................................................... 61
ALLOC3 (nn) ............................................................................................................. 61
Notes ......................................................................................................................... 61
ALLOC4 (nn) ............................................................................................................. 62
ASSOC ...................................................................................................................... 62
Notes ......................................................................................................................... 62
ASSOC (nn) ............................................................................................................. 62
AVLRECL=nnn/KEEP .............................................................................................. 63
File Modelling ........................................................................................................ 63
Note ......................................................................................................................... 63
AVRL ......................................................................................................................... 63
Notes ......................................................................................................................... 63
AVRL (nn) ............................................................................................................. 63
BLKSIZE (nn) ......................................................................................................... 64
Notes ......................................................................................................................... 64
BUFFP (nn) ............................................................................................................ 64
Notes ......................................................................................................................... 64
BUFFP/IXL (nn) .................................................................................................... 64
Notes ......................................................................................................................... 64
CAT=xxx.xx ........................................................................................................... 65
Notes ......................................................................................................................... 65
CATALOG (nn) ....................................................................................................... 65
Notes ......................................................................................................................... 65
CBLCLINE=nnn ....................................................................................................... 65
Note ......................................................................................................................... 66
CBLVCALE=nnn ...................................................................................................... 66
Note ......................................................................................................................... 66
CBLVCALW=nnn ..................................................................................................... 66
Note ......................................................................................................................... 66
CBLVCEXT=nnn ...................................................................................................... 66
Notes ......................................................................................................................... 67
CBLVCFS=xxx ........................................................................................................ 67
CBLVCFS=xxx ........................................................................................................ 67
Notes ......................................................................................................................... 67
CBLVCFCN=xxx ...................................................................................................... 67
Notes ......................................................................................................................... 67
CBLVCN=xxx .......................................................................................................... 68
Notes ......................................................................................................................... 68
CBLVCSCA=nnn ...................................................................................................... 69
Notes ......................................................................................................................... 69
CBLVCSCI=nnn ...................................................................................................... 69
Notes ......................................................................................................................... 69
CBLVCSPA=nnn ...................................................................................................... 69
Notes ......................................................................................................................... 69
CBLVCSP=nnn ......................................................................................................... 70
Notes ......................................................................................................................... 70
CBLVCW1=xx .......................................................................................................... 70
CBLVCW2=xx .......................................................................................................... 70
CBLVCW3=xx .......................................................................................................... 70
CBLVCW4=xx .......................................................................................................... 71
CBLVCW5=xx .......................................................................................................... 71
CBLVCW6=xx .......................................................................................................... 72
CBLVCW7=xx .......................................................................................................... 72
CBLVCW8=xx .......................................................................................................... 72
CBLVCW9=xx .......................................................................................................... 72
CL/CA ....................................................................................................................... 73
Notes ......................................................................................................................... 73
CIMAX=nnnn .......................................................................................................... 74
Notes ......................................................................................................................... 74
CIMIN=nnnn ............................................................................................................ 74
Notes ......................................................................................................................... 74
CISIZE (nn) ............................................................................................................ 74
Notes ......................................................................................................................... 74
CISIZE=nnn ............................................................................................................. 74
Notes ......................................................................................................................... 74
CISIZE=KEEP ......................................................................................................... 75
Notes ......................................................................................................................... 75
CISIZE=n1.n2 ......................................................................................................... 75
Notes ......................................................................................................................... 75
CLASS=n .................................................................................................................. 75
COMPONENT (nn) ................................................................................................ 75
# Contents

**A-Z Reference**

<table>
<thead>
<tr>
<th>Entry</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notes</td>
<td>76</td>
</tr>
<tr>
<td>COMPRESSED.</td>
<td>76</td>
</tr>
<tr>
<td>CREATED (nn).</td>
<td>76</td>
</tr>
<tr>
<td>Notes</td>
<td>76</td>
</tr>
<tr>
<td>CYLIND (nn).</td>
<td>76</td>
</tr>
<tr>
<td>CYLMAX=nnnn</td>
<td>77</td>
</tr>
<tr>
<td>DDNAME=fname...</td>
<td>77</td>
</tr>
<tr>
<td>Notes</td>
<td>77</td>
</tr>
<tr>
<td>DDNAME=fname...</td>
<td>77</td>
</tr>
<tr>
<td>Notes</td>
<td>77</td>
</tr>
<tr>
<td>DEFINE</td>
<td>77</td>
</tr>
<tr>
<td>Notes</td>
<td>77</td>
</tr>
<tr>
<td>DEFINED (nnn)</td>
<td>78</td>
</tr>
<tr>
<td>Notes</td>
<td>78</td>
</tr>
<tr>
<td>DEFINED...</td>
<td>78</td>
</tr>
<tr>
<td>Notes</td>
<td>78</td>
</tr>
<tr>
<td>DEV=</td>
<td>78</td>
</tr>
<tr>
<td>Notes</td>
<td>78</td>
</tr>
<tr>
<td>DEV=ALL</td>
<td>78</td>
</tr>
<tr>
<td>VSE systems...</td>
<td>79</td>
</tr>
<tr>
<td>MVS systems...</td>
<td>79</td>
</tr>
<tr>
<td>DSN=xxx.xxx.xx etc...</td>
<td>80</td>
</tr>
<tr>
<td>Notes</td>
<td>80</td>
</tr>
<tr>
<td>DSN (nn)...</td>
<td>80</td>
</tr>
<tr>
<td>Notes</td>
<td>80</td>
</tr>
<tr>
<td>DSN (nn)...</td>
<td>80</td>
</tr>
<tr>
<td>Notes</td>
<td>81</td>
</tr>
<tr>
<td>EXPORT...</td>
<td>81</td>
</tr>
<tr>
<td>Notes</td>
<td>81</td>
</tr>
<tr>
<td>EXCP...</td>
<td>81</td>
</tr>
<tr>
<td>Notes</td>
<td>82</td>
</tr>
<tr>
<td>EXP=PERM...</td>
<td>82</td>
</tr>
<tr>
<td>Notes</td>
<td>82</td>
</tr>
<tr>
<td>EXP=TEMP...</td>
<td>82</td>
</tr>
<tr>
<td>Notes</td>
<td>83</td>
</tr>
<tr>
<td>EXPD...</td>
<td>83</td>
</tr>
<tr>
<td>Notes</td>
<td>83</td>
</tr>
<tr>
<td>EXPIRES (nn...</td>
<td>84</td>
</tr>
<tr>
<td>REPORT VTOC</td>
<td>84</td>
</tr>
<tr>
<td>Notes</td>
<td>84</td>
</tr>
<tr>
<td>EXPIRES (nn)...</td>
<td>84</td>
</tr>
<tr>
<td>REPORT VCAT</td>
<td>84</td>
</tr>
<tr>
<td>Notes</td>
<td>84</td>
</tr>
<tr>
<td>EXT (nn)...</td>
<td>85</td>
</tr>
<tr>
<td>Notes</td>
<td>85</td>
</tr>
<tr>
<td>EXTADDR...</td>
<td>85</td>
</tr>
<tr>
<td>Notes</td>
<td>85</td>
</tr>
<tr>
<td>EXTENDED...</td>
<td>85</td>
</tr>
<tr>
<td>Notes</td>
<td>85</td>
</tr>
<tr>
<td>EXTNO=nnn...</td>
<td>85</td>
</tr>
<tr>
<td>Notes</td>
<td>85</td>
</tr>
<tr>
<td>FALT...</td>
<td>85</td>
</tr>
<tr>
<td>Notes</td>
<td>86</td>
</tr>
<tr>
<td>FREE...</td>
<td>86</td>
</tr>
<tr>
<td>Notes</td>
<td>86</td>
</tr>
<tr>
<td>FREEBYTES (nn)...</td>
<td>86</td>
</tr>
<tr>
<td>Notes</td>
<td>86</td>
</tr>
<tr>
<td>FREETAB...</td>
<td>86</td>
</tr>
<tr>
<td>Notes</td>
<td>87</td>
</tr>
<tr>
<td>FRSP (nn)...</td>
<td>87</td>
</tr>
<tr>
<td>Notes</td>
<td>87</td>
</tr>
<tr>
<td>FRSPCA=nnn...</td>
<td>88</td>
</tr>
<tr>
<td>Notes</td>
<td>88</td>
</tr>
<tr>
<td>FRSPCI=nnn...</td>
<td>88</td>
</tr>
<tr>
<td>Notes</td>
<td>88</td>
</tr>
<tr>
<td>GDGDSN...</td>
<td>88</td>
</tr>
<tr>
<td>Notes</td>
<td>88</td>
</tr>
<tr>
<td>GDGPT (GDG repeat)...</td>
<td>89</td>
</tr>
<tr>
<td>Notes</td>
<td>89</td>
</tr>
<tr>
<td>GGEN (nn) (GDG gen nos)...</td>
<td>89</td>
</tr>
<tr>
<td>Notes</td>
<td>89</td>
</tr>
<tr>
<td>GMAX (nn) (GDG max level)...</td>
<td>89</td>
</tr>
<tr>
<td>Notes</td>
<td>90</td>
</tr>
<tr>
<td>GROWTH=nnn...</td>
<td>90</td>
</tr>
<tr>
<td>Notes</td>
<td>90</td>
</tr>
<tr>
<td>GVER (nn)...</td>
<td>90</td>
</tr>
<tr>
<td>Notes</td>
<td>90</td>
</tr>
<tr>
<td>HEAD=string...</td>
<td>90</td>
</tr>
<tr>
<td>Notes</td>
<td>90</td>
</tr>
<tr>
<td>HIALRBA (nn)...</td>
<td>90</td>
</tr>
<tr>
<td>Notes</td>
<td>90</td>
</tr>
<tr>
<td>HIBLK=nnnn...</td>
<td>91</td>
</tr>
<tr>
<td>Notes</td>
<td>91</td>
</tr>
<tr>
<td>HICISZ=nnnn...</td>
<td>92</td>
</tr>
<tr>
<td>Notes</td>
<td>92</td>
</tr>
<tr>
<td>HICYL=nnnn...</td>
<td>92</td>
</tr>
<tr>
<td>Notes</td>
<td>92</td>
</tr>
</tbody>
</table>

2013-10-29 15:43:01

iv
## Contents

### A-Z Reference

<table>
<thead>
<tr>
<th>Definition</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIDATE=</td>
<td>92</td>
</tr>
<tr>
<td>Notes</td>
<td>93</td>
</tr>
<tr>
<td>HIEXP=</td>
<td>92</td>
</tr>
<tr>
<td>Notes</td>
<td>93</td>
</tr>
<tr>
<td>HIT=</td>
<td>93</td>
</tr>
<tr>
<td>HIUSERB (nn)</td>
<td>93</td>
</tr>
<tr>
<td>IGN=</td>
<td>93</td>
</tr>
<tr>
<td>Notes</td>
<td>94</td>
</tr>
<tr>
<td>IMB</td>
<td>94</td>
</tr>
<tr>
<td>Notes</td>
<td>94</td>
</tr>
<tr>
<td>INDEX=</td>
<td>94</td>
</tr>
<tr>
<td>INFO (nn)</td>
<td>94</td>
</tr>
<tr>
<td>File Messages</td>
<td>95</td>
</tr>
<tr>
<td>Summary Messages</td>
<td>95</td>
</tr>
<tr>
<td>ISC</td>
<td>95</td>
</tr>
<tr>
<td>Notes</td>
<td>96</td>
</tr>
<tr>
<td>IXL (nn)</td>
<td>96</td>
</tr>
<tr>
<td>Notes</td>
<td>97</td>
</tr>
<tr>
<td>KEY=</td>
<td>96</td>
</tr>
<tr>
<td>Notes</td>
<td>97</td>
</tr>
<tr>
<td>KL (nn)</td>
<td>96</td>
</tr>
<tr>
<td>Notes</td>
<td>97</td>
</tr>
<tr>
<td>KL/BLK/IMB (nn)</td>
<td>97</td>
</tr>
<tr>
<td>Notes</td>
<td>97</td>
</tr>
<tr>
<td>LINESPACE=n</td>
<td>97</td>
</tr>
<tr>
<td>Notes</td>
<td>97</td>
</tr>
<tr>
<td>LIST=YES</td>
<td>97</td>
</tr>
<tr>
<td>Notes</td>
<td>98</td>
</tr>
<tr>
<td>LIST LABL.</td>
<td>98</td>
</tr>
<tr>
<td>Notes</td>
<td>98</td>
</tr>
<tr>
<td>LISTVCAT</td>
<td>98</td>
</tr>
<tr>
<td>LSTVTOG</td>
<td>98</td>
</tr>
<tr>
<td>LMAX</td>
<td>98</td>
</tr>
<tr>
<td>Notes</td>
<td>98</td>
</tr>
<tr>
<td>LMAX (nn)</td>
<td>99</td>
</tr>
<tr>
<td>Notes</td>
<td>99</td>
</tr>
<tr>
<td>LOBLK=nnnn</td>
<td>99</td>
</tr>
<tr>
<td>LOCISZ=nnnn</td>
<td>100</td>
</tr>
<tr>
<td>LOCYL=nnnn</td>
<td>100</td>
</tr>
<tr>
<td>LOVOL=</td>
<td>100</td>
</tr>
<tr>
<td>Notes</td>
<td>100</td>
</tr>
<tr>
<td>LOEXP=</td>
<td>100</td>
</tr>
<tr>
<td>LTRK=nnnn</td>
<td>101</td>
</tr>
<tr>
<td>LRECL (nn)</td>
<td>101</td>
</tr>
<tr>
<td>MAXLRECL=nnnn/KEEP</td>
<td>101</td>
</tr>
<tr>
<td>MOUNT</td>
<td>101</td>
</tr>
<tr>
<td>Notes</td>
<td>102</td>
</tr>
<tr>
<td>NEWDSN=xxx.xxx.xx etc</td>
<td>102</td>
</tr>
<tr>
<td>Notes</td>
<td>102</td>
</tr>
<tr>
<td>NEWVOL=xxx</td>
<td>102</td>
</tr>
<tr>
<td>Notes</td>
<td>102</td>
</tr>
<tr>
<td>NOALIAS</td>
<td>103</td>
</tr>
<tr>
<td>NOASSOC</td>
<td>103</td>
</tr>
<tr>
<td>NODEFINE</td>
<td>103</td>
</tr>
<tr>
<td>NOEXPD</td>
<td>103</td>
</tr>
<tr>
<td>NOFREE</td>
<td>103</td>
</tr>
<tr>
<td>NOGDDSN</td>
<td>104</td>
</tr>
<tr>
<td>NOGDRPT</td>
<td>104</td>
</tr>
<tr>
<td>NOINDEX</td>
<td>104</td>
</tr>
<tr>
<td>Notes</td>
<td>104</td>
</tr>
<tr>
<td>NOISCC</td>
<td>104</td>
</tr>
<tr>
<td>NOMOUNT</td>
<td>104</td>
</tr>
<tr>
<td>Notes</td>
<td>104</td>
</tr>
<tr>
<td>NOOVLAY</td>
<td>105</td>
</tr>
<tr>
<td>NOPCAP</td>
<td>105</td>
</tr>
<tr>
<td>NOPERM</td>
<td>105</td>
</tr>
<tr>
<td>Notes</td>
<td>105</td>
</tr>
<tr>
<td>NOPJCL</td>
<td>105</td>
</tr>
<tr>
<td>NOPSEV</td>
<td>105</td>
</tr>
<tr>
<td>NOVOL</td>
<td>106</td>
</tr>
<tr>
<td>Notes</td>
<td>106</td>
</tr>
<tr>
<td>NRECS (nn)</td>
<td>106</td>
</tr>
<tr>
<td>Notes</td>
<td>106</td>
</tr>
<tr>
<td>NRECS=nnnn</td>
<td>106</td>
</tr>
</tbody>
</table>
# Contents

**A-Z Reference**

<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSEG (nn)</td>
<td>106</td>
</tr>
<tr>
<td>Notes</td>
<td>106</td>
</tr>
<tr>
<td>OPTION</td>
<td>107</td>
</tr>
<tr>
<td>Notes</td>
<td>107</td>
</tr>
<tr>
<td>OVLAY</td>
<td>107</td>
</tr>
<tr>
<td>Notes</td>
<td>107</td>
</tr>
<tr>
<td>OWN=xx</td>
<td>107</td>
</tr>
<tr>
<td>Notes</td>
<td>107</td>
</tr>
<tr>
<td>PAGEDEPTH=nn</td>
<td>108</td>
</tr>
<tr>
<td>Notes</td>
<td>108</td>
</tr>
<tr>
<td>PAGEWIDTH=nn</td>
<td>108</td>
</tr>
<tr>
<td>Notes</td>
<td>108</td>
</tr>
<tr>
<td>PAIN=passed</td>
<td>108</td>
</tr>
<tr>
<td>Notes</td>
<td>108</td>
</tr>
<tr>
<td>PCNT</td>
<td>108</td>
</tr>
<tr>
<td>PCNT (nn)</td>
<td>109</td>
</tr>
<tr>
<td>Notes</td>
<td>109</td>
</tr>
<tr>
<td>PERM</td>
<td>109</td>
</tr>
<tr>
<td>Notes</td>
<td>109</td>
</tr>
<tr>
<td>PHYREG (nn)</td>
<td>109</td>
</tr>
<tr>
<td>Notes</td>
<td>109</td>
</tr>
<tr>
<td>PRTCAP</td>
<td>110</td>
</tr>
<tr>
<td>Notes</td>
<td>110</td>
</tr>
<tr>
<td>PRTJCL</td>
<td>110</td>
</tr>
<tr>
<td>Notes</td>
<td>110</td>
</tr>
<tr>
<td>PRTSEV</td>
<td>110</td>
</tr>
<tr>
<td>Notes</td>
<td>110</td>
</tr>
<tr>
<td>QUERY</td>
<td>110</td>
</tr>
<tr>
<td>Notes</td>
<td>110</td>
</tr>
<tr>
<td>RAW=foo</td>
<td>110</td>
</tr>
<tr>
<td>Notes</td>
<td>110</td>
</tr>
<tr>
<td>RECDEL (nn)</td>
<td>112</td>
</tr>
<tr>
<td>Notes</td>
<td>112</td>
</tr>
<tr>
<td>RECFM (nn)</td>
<td>112</td>
</tr>
<tr>
<td>Notes</td>
<td>112</td>
</tr>
<tr>
<td>RECIIP (nn)</td>
<td>112</td>
</tr>
<tr>
<td>Notes</td>
<td>112</td>
</tr>
<tr>
<td>RECIINS (nn)</td>
<td>113</td>
</tr>
<tr>
<td>Notes</td>
<td>113</td>
</tr>
<tr>
<td>RECORDS=nnn</td>
<td>113</td>
</tr>
<tr>
<td>Notes</td>
<td>113</td>
</tr>
<tr>
<td>RECSTATS (nn)</td>
<td>113</td>
</tr>
<tr>
<td>Notes</td>
<td>113</td>
</tr>
<tr>
<td>RECPD (nn)</td>
<td>113</td>
</tr>
<tr>
<td>Notes</td>
<td>113</td>
</tr>
<tr>
<td>REF=xxx.xx</td>
<td>113</td>
</tr>
<tr>
<td>Notes</td>
<td>113</td>
</tr>
<tr>
<td>REF=ALL</td>
<td>114</td>
</tr>
<tr>
<td>Notes</td>
<td>114</td>
</tr>
<tr>
<td>REPORT</td>
<td>114</td>
</tr>
<tr>
<td>Notes</td>
<td>114</td>
</tr>
<tr>
<td>RKP (nn)</td>
<td>115</td>
</tr>
<tr>
<td>Notes</td>
<td>115</td>
</tr>
<tr>
<td>S/C</td>
<td>115</td>
</tr>
<tr>
<td>Notes</td>
<td>115</td>
</tr>
<tr>
<td>S/C (nn)</td>
<td>115</td>
</tr>
<tr>
<td>Notes</td>
<td>115</td>
</tr>
<tr>
<td>SECAJLOG=nn</td>
<td>115</td>
</tr>
<tr>
<td>Notes</td>
<td>115</td>
</tr>
<tr>
<td>SEV=n</td>
<td>116</td>
</tr>
<tr>
<td>Notes</td>
<td>116</td>
</tr>
<tr>
<td>SEVL (nn)</td>
<td>116</td>
</tr>
<tr>
<td>Notes</td>
<td>116</td>
</tr>
<tr>
<td>SHR</td>
<td>116</td>
</tr>
<tr>
<td>Notes</td>
<td>116</td>
</tr>
<tr>
<td>SHR (nn)</td>
<td>116</td>
</tr>
<tr>
<td>Notes</td>
<td>116</td>
</tr>
<tr>
<td>SMS (nn)</td>
<td>116</td>
</tr>
<tr>
<td>Notes</td>
<td>116</td>
</tr>
<tr>
<td>SMSMD (nn)</td>
<td>117</td>
</tr>
<tr>
<td>Notes</td>
<td>117</td>
</tr>
<tr>
<td>SMSIM (nn)</td>
<td>117</td>
</tr>
<tr>
<td>Notes</td>
<td>117</td>
</tr>
<tr>
<td>SMSN (nn)</td>
<td>117</td>
</tr>
<tr>
<td>Notes</td>
<td>117</td>
</tr>
<tr>
<td>SORT=xxx</td>
<td>118</td>
</tr>
<tr>
<td>Notes</td>
<td>118</td>
</tr>
<tr>
<td>SORT</td>
<td>118</td>
</tr>
<tr>
<td>Notes</td>
<td>118</td>
</tr>
<tr>
<td>SORTD</td>
<td>118</td>
</tr>
<tr>
<td>SPANDED</td>
<td>118</td>
</tr>
<tr>
<td>SPLIT=nn</td>
<td>118</td>
</tr>
<tr>
<td>Notes</td>
<td>118</td>
</tr>
<tr>
<td>SPLITCA (nn)</td>
<td>118</td>
</tr>
<tr>
<td>Notes</td>
<td>118</td>
</tr>
<tr>
<td>SPLITCI (nn)</td>
<td>119</td>
</tr>
<tr>
<td>Notes</td>
<td>119</td>
</tr>
<tr>
<td>SPLITSTATS</td>
<td>119</td>
</tr>
<tr>
<td>Notes</td>
<td>119</td>
</tr>
<tr>
<td>START (nn)</td>
<td>119</td>
</tr>
<tr>
<td>STOPAFT=nnn</td>
<td>119</td>
</tr>
<tr>
<td>Notes</td>
<td>119</td>
</tr>
</tbody>
</table>
# Contents

## Messages

- Console Messages .................................................................................................................................................. 155
- Expiry Warning .................................................................................................................................................. 155
- Error Messages .................................................................................................................................................. 156
- Wrong Volume .................................................................................................................................................. 156
- Miscellaneous Messages ................................................................................................................................. 156
- Return Codes .................................................................................................................................................. 156

## Technical Information

- Installation .......................................................................................................................................................... 158
- Introduction ........................................................................................................................................................ 158
- The Distribution Material ................................................................................................................................. 158
- The Operating Environment ............................................................................................................................. 158
- System Components ...................................................................................................................................... 158
- CBLVCAT Build Level Information .................................................................................................................. 159
- System Considerations .................................................................................................................................... 159
- Year 2000 Compliance .................................................................................................................................. 159
- Tuning Principles .............................................................................................................................................. 160
  - Approach to ALLOCATION ............................................................................................................................ 160
  - Approach to RECORDSIZE ............................................................................................................................ 161
  - Approach to DATA CISIZE ............................................................................................................................... 161
  - Approach to INDEX CISIZE ........................................................................................................................... 161
  - Approach to IMBED/REPLICATE ...................................................................................................................... 161
  - Approach to SPEED v RECOVERY ................................................................................................................ 161
  - Approach to WRITECHECK ............................................................................................................................ 162
  - Approach to SHAREOPTION .......................................................................................................................... 162
  - Approach to SPANNED .................................................................................................................................. 162
  - Approach to FREESPACE .............................................................................................................................. 162
  - Approach to BUFFERSPACE ............................................................................................................................ 162
- Tuning for IMS/DL1 Databases .......................................................................................................................... 163
  - IMS/DL1 Data Sets ....................................................................................................................................... 164
  - IMS/DL1 Database Processing ......................................................................................................................... 164
  - Database Re-organisation ............................................................................................................................... 164
  - Buffer Pool Allocation .................................................................................................................................. 164
- Additional VSAM Information ............................................................................................................................ 165
  - MIN-CA and MAX-CA .................................................................................................................................... 165
  - CASIZE ............................................................................................................................................................ 166
  - CISIZE .............................................................................................................................................................. 166
  - SECONDARY ALLOCATION ............................................................................................................................ 167
  - IMBED v REPLICATE ..................................................................................................................................... 168
  - FREESPACE .................................................................................................................................................... 168
  - SPLITS ............................................................................................................................................................. 169
  - KEY COMPRESSION ....................................................................................................................................... 170
  - ALTERNATE INDEX and PATH .......................................................................................................................... 170
  - LEVELS OF INDEX ......................................................................................................................................... 171
  - BUFFER SPACE ............................................................................................................................................. 172
  - SPEED v RECOVERY .................................................................................................................................... 174

## Glossary of Terms

- ................................................................. 175

## Also from CBL

- SELCOPY the Productivity Aid .............................................................................................................................. 180
**Documentation Notes**


**Copyright** in the whole and every part of this document and of the CBLVCAT system and programs is owned by Compute (Bridgend) Ltd, whose registered office is located at 8 Merthyr Mawr Road, Bridgend, Wales, UK, CF31 3NH, and who reserve the right to alter at their convenience the whole or any part of this document, or the CBLVCAT system and programs.

No reproduction of the whole or any part of the CBLVCAT system and programs, or of this document, is to be made without prior written authority from Compute (Bridgend) Ltd.

At the time of publication, this document is believed to be correct. Where the program product differs from that stated herein, Compute (Bridgend) Ltd reserve the right to revise either the program or its documentation at their discretion.

CBL do not warrant that upward compatibility will be maintained for any use made of this program product to perform any operation in a manner not documented within the user manual.
Summary of Changes

This section summarises those new features and changes included in CBLVCAT Release 2.12, which have incorporated into the body of this manual.


Note that changes and new features may be subject to alteration, at the discretion of CBL, according to advice and feedback from users. CBL welcomes your new feature suggestions and requirements.

Introduced in Rel 9.8 (April 1997)

- Year 2000 Compliance resulting in changes to the following:
  - **All**: Standard Catalog Report layout.
  - **All**: Standard VTOC Report layout.
  - **All**: REPORT VCAT and REPORT VTOC fields.
  - **All**: LISTVCAT DD=ucat OPTION DEFINE.
  - **All**: VSE: MOD DEV=cuu DSN=xxx.. EXP=PERM.
  - **VSE**: MOD DEV=cuu DSN=xxx.. EXP=TEMP.

- **MVS**: CSA Storage above the line for ICF Catalogs.
- **All**: VSAM (non-ICF) LISTVCAT In-Storage Catalog (ISC) for Improved Performance.
- **All**: Local TIMESTMP for ICF and VSAM Catalog Reports.
- **All**: Variable-Length RRDS (VRDS) reporting.
- **All**: SUBSET TYPE=V and TYPE=M for VRDS selection.
- **ICF**: LISTVCAT VVDS=volser non-VSAM data set reporting.

Introduced in Rel 2.00 (May 2001)

- **All**: CBLNAME Extensions for CBLVCAT Date Range(s) & Password.
- **All**: EXPIRES (nn) for REPORT VCAT.
- **All**: UNIT (nn) for REPORT VTOC.

Introduced in Rel 2.10 (February 2004)

- **All**: 31-bit Addressability.
- **All**: CBLVCAT Build Level & Command -V for SYSLOG/SYSLST Display.
- **All**: Support for File Sizes > 4GB.
- **All**: LISTVCAT SUBSET Parameters
  - **All**: REPORT VCAT EXT
  - **ALL**: LISTVCAT CBLVCSW9=X'10' for TYPE Entries
  - **ALL**: LISTVCAT CBLVCSW9=X'08' for Hex DEVICE Type
  - **MVS**: LISTVTOC SUBSET TYPE=PDSE
  - **MVS**: LISTVTOC REPORT TYPE: PDSE
  - **MVS**: LISTVCAT KEY=xxx.xx Implied REF=xxx.xx.
  - **VSE**: LISTVCAT CAT=xxx.xx & Dynamic Label.
  - **All**: OPTION RAW=fname for all REPORT Fields.
  - **VSE**: LISTLABL SYSNO & VOLUME Entries: *NO*
  - **All**: LISTVCAT DEFINE parameters: MVS SMS Classes & VSE EXTRALARGEDATASET.
  - **All**: OPT - abbreviation for OPTIONS.
  - **All**: TY - abbreviation for TYPE.

Introduced in Rel 2.12 (January 2007)

- **All**: CBLVCAT Interactive Reports via the SELCOPY/i Interactive Environment.
Introduction

General

CBLVCAT is an indispensible utility for all VSAM users. Its Catalog and VTOC reports are clear, concise and legible. These reports highlight problem files and optionally contain tuning recommendations. This insulates you from the volumes of IDCAMS LISTCAT reports and hours of research and calculations which would otherwise be required to accomplish these tasks. In other words, CBLVCAT is your VSAM expert, or your VSAM experts servant. Here is what CBLVCAT will do for you:

Report

1. ICF and VSAM Catalogs.
   You can display whole Catalogs, or just the files of interest. If the standard output does not suit, customise the report to display the required information. It is possible to report on multiple Catalogs in the same run and even produce a variety of reports of the same Catalog. Customised reports can be combined, even with VTOC reports, into a single listing and/or sorted. It is also possible, for ICF Catalogs, to report directly from a VVDS.

2. VTOCs.
   Display your VTOCs in a very legible report. You can choose the sequence and/or a number of other options which let you control the report contents, including report merging.

3. Label Information Area.
   VSE users can display their Partition, Class and Standard Labels in a concise readable report.

Tune

CBLVCAT optionally gives tuning recommendations for problem files. Optimally tuned files result in large savings in processing power, elapsed run time, response time and disk space. It can also tune files based on different operating systems or DASD. This makes it ideal for planning and facilitating a migration.

Monitor

Your periodic reports may be restricted to files that need attention. These reports will save your time, because you won't be dealing with mounds of paper and/or masses of irrelevant information.

IDCAMS DEFINE Parameters

You can optionally punch card images of a DELETE/DEFINE deck, with or without tuning recommendations.

VTOC Modification

If you run VSE you can modify file expiry dates and rename, or delete, files.

The CBLVCAT Approach

The philosophy behind the CBLVCAT approach is simplicity:

Installs Easily

If you are a first time user, just follow the Installation Guide for your type of system and you will be operational very quickly. Later, you can tailor the CBLNAME module to suit your environment and select default preferences for reporting. If you are already a user, or if you use another CBL product, installation is even faster.

No Hooks into Your System

CBLVCAT is risk free. It will not require you to make any system software changes and won't hinder any future software upgrades.

Easy to Use

The control records, used to direct CBLVCAT, are free-format. You can be as general, or specific, as you like, when selecting the files to appear on your reports. A standard report format is provided, which contains the most frequently used information. If you prefer, you can create your own report format, choosing both the fields and the order in which they are displayed.

You Still Control Your Own System

Tuning recommendations and warning indicators are given, but no changes are made behind the scenes.

Speed of Execution

CBLVCAT is written in Assembler Language and reads Catalogs directly. Its speed is astonishing, allowing reports to be produced in prime-time.

Portable

CBLVCAT is not operating system dependent. If you decide to change, simply re-install it on the new system.
The New User

Job Control Language
You do, of course, have to supply **Job Control Language (JCL)** statements, to both invoke CBLVCAT and to link the files, mentioned in the CBLVCAT control statements, with real physical data stored on a computer readable medium. This JCL is operating system dependent and is described later. However, the CBLVCAT control statements used in this section are all operating system independent.

Reports
A standard report of the default Catalog can be produced by simply using the following CBLVCAT command:

```cblvcat```

A report of a specific Catalog is produced with the following:

```cblvcat DDNAME=mycat```

For a standard VTOC report of the volume with VOLID abcvol, use:

```listvtoc VOL=abcvol```

In both types of report any file attributes needing attention are highlighted (either with asterisks or plus signs).

For VSE users, a standard report of the Label Information Area is produced as follows:

```listlabl```

Abbreviations
Most CBLVCAT commands can be abbreviated to save time and space (see Summary of Syntax for a full list). e.g.

- `LC DD=mycat` * Abbreviation for LISTVCAT.
- `LV VOL=abcvol` * Abbreviation for LISTVTOC.

Selective Reporting
So far, the reports produced would have contained all the files within a particular Catalog/Volume. Most of the time you will probably only be interested in a particular file, or set of files. The **SUBSET** parameter, together with its associated sub-parameters, designates which files appear in the report (see Summary of Syntax for sub-parameter details). e.g.

```lc DD=mycat SUBSET TYPE=K```

Multiple SUBSET sub-parameters are allowed. They are processed as a logical AND i.e. cumulative (KEY and IGN are exceptions, being processed as a logical OR.)

The **SUBSET** parameter is only effective for the operation on which it is coded and the word **SUBSET** is optional.

Tuning
If the **TUNE** parameter is specified, CBLVCAT will add its tuning recommendations to the Catalog report. CBLVCAT doesn't make the changes, it lists the IDCAMS DEFINE parameters requiring modification and comments on the changes required. e.g.

```lc DD=mycat TYPE=K TUNE```

The **TUNE** parameter is only effective for the LISTVCAT command on which it is coded.

Report Customisation
If the standard report layout and/or contents don't suit your requirements, you can customise the output. This is accomplished using the **OPTIONS** or **REPORT** commands, together with their associated parameters. Multiple parameters are allowed (see Summary of Syntax for parameter details). The selected **OPTIONS**/ **REPORT** parameters remain in force for the rest of the run, or until reset by another **OPTIONS**/ **REPORT** parameter.

- **OPTIONS** alters the content of the standard report. e.g.

  ```options LMAX```

  This would create a report with maximum record length **LMAX** displayed in place of the average record length **AVRL**.

  **OPTION** can also be supplied as a parameter to a LISTVCAT or LISTVTOC command (see **OPTION** in the A-Z Reference).

- **REPORT** creates a totally customised report. Each parameter specifies a column to appear in the report. e.g.

  ```report VCAT DSN TYPE NRECS```

  This would produce a customised Catalog report containing headings in the order specified, i.e. **DSN** (Data Set Name), **TYPE** File Type (e.g. KSDS) and **NRECS** the number of records in that file.
The MERGE and SORT parameters may be used, in conjunction with the REPORT command, to combine multiple listing into one report and to sort the output. (See Guide to List Output for examples of report customisation).

MVS Execution

Catalog Reports
Under MVS batch, CBLVCAT has two methods for producing a Catalog report (See also TSO execution). These are as follows:

1. Reference the required Catalog using a DDNAME, e.g.

   //STEP1 EXEC PGM=CBLV
   //UCAT DD DSN=vsam.user.cat,DISP=SHR
   //SYSPRINT DD SYSOUT="
   //SYSIN DD *
   LC DD=UCAT TYPE=K TUNE

2. Reference the required Catalog using the REF parameter (REFerence). The argument of this parameter can be the Catalog itself (using the self-defining entry), its alias, or any DSN in the Catalog. If REF=ALL (ICF with DFP 3.1 or higher) is used, a report is produced for all Catalogs. e.g.

   //STEP1 EXEC PGM=CBLV
   //SYSPRINT DD SYSOUT="
   //SYSIN DD *
   LC REF=any.file.in.cat TYPE=K TUNE DEV=3390

Caution
Pre DFP 3.1 users should supply a STEPCAT/JOBCAT to process ICF Catalogs. Contact CBL should this prove to be inconvenient.

For processing non-ICF VSAM Catalogs a STEPCAT/JOBCAT must be supplied. In addition, before running CBLVCAT for the first time on these Catalogs, it is advisable to run IDCAMS VERIFY first, otherwise, CBLVCAT may not be able to report on all files.

VTOC Reports
Under MVS batch, CBLVCAT has three methods for producing a VTOC report. These are as follows:

1. Reference the required Volume using a DDNAME, e.g.

   //STEP1 EXEC PGM=CBLV
   //VOL1 DD VOL=SER=volser,UNIT=3390,DISP=SHR
   //SYSPRINT DD SYSOUT="
   //SYSIN DD *
   LV DD=VOL1

2. Reference the required Volume using the VOL parameter with the required Volume Serial number as the argument. The argument of the VOL parameter can also be a generic subset. For instance, VOL=ABC* will produce VTOC information for all volumes starting with ABC). e.g.

   //STEP1 EXEC PGM=CBLV
   //SYSPRINT DD SYSOUT="
   //SYSIN DD *
   LV VOL=volser

3. Reference the required Volume using the DEV parameter. The argument of this parameter is a user-defined esoteric group of units (generic subset). If DEV=ALL is used, a report is produced for all disk devices identified within SYSALLDA, e.g.

   //STEP1 EXEC PGM=CBLV
   //SYSPRINT DD SYSOUT="
   //SYSIN DD *
   LV DEV=3380DASD

Input via the PARM Field
For both Catalog and VTOC reports, CBLVCAT can take its input via the PARM field of the EXEC statement. e.g.

   //STEP1 EXEC PGM=CBLV,PARM='LV VOL=volser'
   //SYSPRINT DD SYSOUT="

TSO Execution
To simplify the production of Catalog/VTOC reports, the V TSO REXX exec is provided. It is supplied on the distribution tape and will have been made available to you as part of the installation procedure (see separate CBLVCAT installation guide).

The control statements for producing Catalog and VTOC reports are described under their relevant headings, and the various methods of supplying these statements to V TSO are described below under the heading "Invoking V TSO".
Catalog Reports

Authorisation to read a Catalog, needs to be set via an AUTHCMD entry within module IKJTSO00.

Under TSO, CBLVCAT has two methods for producing a Catalog report as follows:

1. Reference the required Catalog using the DDNAME, e.g. parameter (This method requires an ALLOC statement to have been issued for the required Catalog). The CBLVCAT control statement uses the ALLOC name as the argument of the DDNAME parameter.
   
   \[ LC \text{ DD=UCAT TYPE=K TUNE} \]

2. Reference the required Catalog using the REF parameter. The argument of this parameter can be the DSN of the Catalog itself (using the self-defining entry), its alias, or any DSN in the Catalog. If REF=ALL (ICF with DFP 3.1 or higher) is used, a report is produced for all Catalogs.
   
   \[ LC \text{ REF=any.file.in.catalog} \]

VTOC Reports

Under TSO, CBLVCAT has three methods for producing a VTOC report. These are as follows:

1. Reference the required Volume using the DDNAME parameter (This method requires an ALLOC statement to have been issued for the required Volume). The CBLVCAT control statement uses the ALLOC name as the argument of the DDNAME parameter.
   
   \[ LV \text{ DD=VOL1} \]

2. Reference the required Volume using the VOL parameter with the required Volume Serial number as the argument. The argument can also be a generic subset (for instance, \[ VOL=ABC* \] will produce VTOC information for all volumes starting with ABC).
   
   \[ LV \text{ VOL=volser} \]

3. Reference the required Volume using the DEV parameter. The argument of this parameter is a user-defined esoteric group of units (generic subset). If DEV=ALL is used, a report is produced for all disk devices identified within SYSALLDA.
   
   \[ LV \text{ DEV=3380DASD} \]

Invoking V TSO

Before invoking V TSO, it is necessary to tailor it to include the name of the PDS to which you require output to be directed (The line to tailor is indicated within the exec). The methods of invoking V TSO are as follows:

1. V
   
   If the exec is invoked with no parameters, CBLVCAT takes its control records from the terminal and directs its output to the PDS member 'VLST' (The PDS name should have already been tailored in the exec).

2. V inarg
   
   The first parameter supplied to V TSO is the input location. This can either be 'TERM' (default), where control records are input from the terminal, or a PDS member which contains control records (The PDS name should have already been tailored in the exec).

3. V inarg outarg
   
   The second parameter supplied to V TSO is the output location (the input parameter must also have been supplied, or ' ' entered to use the default). This can either be 'TERM', where output is directed to the terminal, or the PDS member name of your required output file (The PDS name should have already been tailored in the exec). If this parameter is omitted, output defaults to the PDS member 'VLST'.

4. V (inarg) DEST=xyz SYSOUT=x
   
   The DEST= and/or SYSOUT= parameters can be used to control the output location. They are not positional but must be coded with their arguments following. They cannot be used if the output argument has been specified.

VSE Execution

Catalog Reports

As with all VSAM processing, if a label definition (DLBL) for IJSYSCT does not exist in the System Standard Label Area, one must be supplied as a User-label.

Under VSE/SP 2.1 or higher, all that is required is to point CBLVCAT to any existing Standard or User DLBL which refers to the required catalog.

\[
// \text{EXEC CBLV} \\
\text{LC DD=UCATWK1} \quad * \text{Standard label assumed.} \\
//
\]

\[
// \text{EXEC CBLV} \\
\text{LC DD=MYCAT} \\
//
\]

or

\[
// \text{DLBL MYCAT,'any.user.catalog',,VSAM} \\
// \text{EXEC CBLV} \\
\text{LC DD=MYCAT} \\
//
\]
Under earlier releases of VSE, the DLBL must contain a **CAT** operand with the argument equal to the DLBL fname. e.g.

```
// DLBL CBLCAT,'my.user.cat',,VSAM,CAT=CBLCAT
// EXEC CBLV
LC DD=CBLCAT
/*
```

**VTOC Reports**

Under VSE, CBLVCAT has three methods for producing a VTOC report. These are as follows:

1. Reference the required Volume indirectly using the **SYS** parameter with a temporary or permanently assigned programmer logical unit number as the argument. If **SYS=ALL** is used, a report is produced for all currently assigned disk devices.

2. Reference the required Volume indirectly using the **DEV** parameter with the required device number as the argument. If **DEV=ALL** is used, a report is produced for all active disk devices.

3. Reference the required Volume directly using the **VOL** parameter with the required Volume Serial Number as the argument. The argument of the **VOL** parameter can also be a generic subset. For instance, **VOL=ABC** will produce VTOC information for all volumes starting with ABC. e.g.

```
// EXEC CBLV
LV SY=3               * Assignment to SYS003 required.
LV DEV=141            * Volume on device 141.
LV VOL=DOSRES         * Volume DOSRES.
/*
```

**Standard Label Reports**

Partition (temporary and permanent), Class and System labels are listed with one command. e.g.

```
LISTLABL    * No parameters needed
```

**Input via the PARM Field**

For both Catalog and VTOC reports, CBLVCAT can take its input via the **PARM** field of the EXEC statement. e.g. (Std label for IJSYSCT exists.)

```
// EXEC CBLV,PARM='LC DD=IJSYSCT'
```

**Operator Console Execution**

One approach is to gain access to a partition, by releasing a **PAUSE** job (e.g. R RDR,PAUSEBG) into the relevant partition. A DLBL/ASSGN for the required Catalog/Volume should then be supplied and CBLVCAT invoked as follows:

```
// EXEC CBLV
CBLVCAT recognises that it has been initiated from the console and prompts you to enter control records. At this point, you reply with the required operation. e.g.

LC DD=IJSYSUC
```

When there are no more control records to enter, reply by entering a "/*/", or a null reply ("end-of-block" or ENTER).

**PLEASE NOTE**

If you are running VTOC reports from the console and the operation fails, the default CANCEL action of the **FAIL** parameter is changed to **IGNORE**.

---

**CMS Execution**

To simplify the production of Catalog/VTOC reports for your attached VSE mini-disks, the **V EXEC** is provided. It is supplied as part of the CBL products distribution material and will have been made available to you as part of the installation procedure (see the "CBL Software Install Guide for VM/CMS and VM/VSE Systems").

In order to generate Catalog reports from CMS, IBM product "VSE/VSAM for VM" must be installed.

As with all VSAM processing, if a label definition (DLBL) for the Master Catalog (IJSYSCT) doesn't exist, one must be supplied and the disk on which it resides must be accessed. It is then necessary to supply similar information for all required disks and Catalogs. e.g. Assuming the Master Catalog is on Volume 140 and the required Catalog and VTOC are on Volume 141.
As can be seen above, when the CAT operand is specified, it is mandatory that the DLBL which is referenced by the argument of that CAT operand should have been previously defined. The argument of the CAT operand must be the same as the DLBL fname. There are only two exceptions to this rule:

1. If IJSYSUC (Job Catalog) is used as the DLBL fname, the CAT operand can be omitted as its default is IJSYSUC.
2. If IJSYSCT (Master Catalog) is used as the DLBL fname and no Job Catalog exists, the CAT operand may be omitted as, in this case, its default becomes IJSYSCT.

PLEASE NOTE
If IJSYSCT is used as the DLBL fname and a Job Catalog (IJSYSUC) already exists, the CAT operand must be specified with the argument IJSYSCT.

The control statements for producing Catalog and VTOC reports are described under their relevant headings, and the various methods of supplying these statements to V EXEC are described below under the heading "Invoking V EXEC".

Catalog Reports
Under CMS, a CBLVCAT control record references a Catalog using the DDNAME parameter. The argument of this parameter is the fname specified in the DLBL, e.g. Referencing the Catalog defined on the DLBL with fname CBLCAT (as defined above).

```
LC   DD=CBLCAT
```

VTOC Reports
Under CMS, CBLVCAT produces a VTOC report using the SYS parameter. The argument of the SYS parameter is the logical unit number specified on the ASSGN statement. e.g. Referencing the Volume defined on the ASSGN statement as SYS001 (as defined above).

```
LV   SYS=001
```

Invoking V EXEC
The V EXEC always runs with DOS ON, with 3 different methods of invoking it:

1. EXEC V
   If the user simply invokes the EXEC with no parameters, CBLVCAT control records are entered on the control line after VM READ is displayed.
   Output is to V LISTING A. Thus:

   ```
   EXEC V   #LC DD=IJSYSCT   #LV SYS=001#
   ```
   executes the parameters supplied between the line end characters ("#”).

2. EXEC V fn
   The parameters are executed from the user control file fn CTL *. Output is to fn LISTING A.

3. EXEC V fn ft fm
   The parameters are executed from the user control file fn ft fm. Output is to fn LISTING A.

Interactive Execution
In addition to standard batch execution, CBLVCAT may be executed interactively on any of the mainframe platforms on which CBLVCAT is supported using the SELCOPY/i Interactive Environment.

SELCOPY/i is provided in the CBL Software Products bundle for each operating system and is installed and configured together with CBLVCAT as part of the standard installation procedure. zSeries machines on which CBLVCAT is licensed, are also licensed to run SELCOPY/i and so should be made available to all users.
The CBLVCAT Interactive (VCI) component of SELCOPY/i allows interactive execution of CBLVCAT control statements sourced from a data set or via a command line. The generated report is stored in internal buffers and presented to the user in a window area with coloured highlighting.

The report window supports prefix area commands for each list entry allowing subsequent copy, edit, delete, IDCAMS LISCAT, etc. of a file in CBLVCAT the report output. The report itself may be also edited and optionally saved to an MVS data set, VSE LIBR member ar CMS file.

In addition to the traditional CBLVCAT report, a list window may be generated containing all of the LISTVCAT/LISTVTOC REPORT fields available for customised report output.

Where LISTVCAT option DEFINE is specified, an edit window is automatically opened for the CBLVCAT generated IDCAMS DEFINE job so allowing alteration by the user before it is submitted to batch.

In addition to CBLVCAT execution, SELCOPY/i supports many other tools and facilities, such as a function rich text editor, file search tools and list windows, thus providing users with a powerful working environment.

SELCOPY/i and CBLVCAT Interactive are documented at length in the "SELCOPY/i Reference and User Guide" and "CBL/e Text Editor" manuals which are freely available from the CBL web site.

Figure 1. SELCOPY/i CBLVCAT Interactive Window & DEFINE Output.
Control Card Syntax Rules

1. If input is via the PARM field of the EXEC statement, then SYSIN/SYSIPT is ignored.

2. If input is via SYSIN/SYSIPT, columns 1 to 71 are used by CBLVCAT and may contain data or comment, column 72 must be left blank and columns 73 to 80 may contain sequence numbers, or any other data the user wishes. CBLVCAT’s commands and/or parameters may contain any number of intervening blanks.

3. If input is via in-stream SYSIN/SYSIPT, users of VSE, CMS/DOS, TSO, and CMS (VM/ESA 1.2.1 or later) must code a /* control statement to signify end of control card statements. Users of CMS (VM/ESA 1.2 or earlier) may use a /* or a null line, whereas native MVS users may omit this statement altogether.

4. If not already processed by the operating system, a /* in positions 1 and 2, followed by a blank, is taken as End-of-File by CBLVCAT, regardless of the rest of the record and regardless of the operating system. CBLVCAT’s control card input file, SYSIN for MVS and CMS, SYSIPT for DOS, is then closed, without attempting to read any further control cards.

5. It is sometimes irritating to have to use a whole record for a single control statement. To overcome this, a separator character may be used to split a control record into logical control records (A separator character immediately followed by /* indicates End-of-File). The default separator character is the "Exclamation Mark" (X'5A'), but this default may be modified by an entry in the CBLNAME module (See that chapter for more details).

6. A CBLVCAT operation consists of a command followed by any number of associated parameters and sub-parameters (See Summary of Syntax for a full list).
   ♦ LISTVCAT and LISTVTOC are examples of commands.
   ♦ DDNAME and REF are examples of parameters.
   ♦ LMAX and TYPE are examples of sub-parameters.
   A command must start on a new logical record. The parameters and sub-parameters which follow, belong to the operation started by that command. An operation is completed as soon as the next command, or end of file on SYSIN/SYSIPT, is detected. There can be more than one command per execution.

   Commands and parameters may be supplied in mixed case characters. CBLVCAT converts all input to upper case.

7. There is no continuation character. Commands and parameters may be specified on more than one record (The two exceptions are the LISTLABL and QUERY commands, which must be complete on one logical record), but commands, parameters and strings cannot be broken in the middle.

8. Comments are allowed. An asterisk, which is not part of a quoted literal string, signals the start of a comment. All data following the asterisk is treated as comment. An entire record may be used as a comment line by coding an asterisk as the first non-blank character, or the entire control record may be left blank.

9. Some parameters are in KEYWORD=string format. CBLVCAT treats the equal-sign as a blank, which allows the substitution of one or more blanks in its place.

   If a string contains blanks, commas or asterisks, it must be enclosed in single quotes. If it includes a quote, the quote must be represented as two quotes. e.g.

   MOD DSN='FILE ID' NEWDSN=''A'' FILE'
Guide to List Output

This chapter illustrates and explains the output from CBLVCAT if the TUNE parameter is not specified. (See also Guide to VSAM Tuning). However, TUNE is mentioned in section IDCAMS DEFINE and Reorganisation) of this chapter to illustrate its use in conjunction with DEFINE.

Chapter Summary of Syntax contains a full list of all functions and their associated parameters and sub-parameters.

Detailed descriptions of functions, parameters and sub-parameters in alphabetical order, are available in chapter A-Z Reference

---

Standard Reports

1. Introduction
2. Example 1 - Standard Catalog Report
3. Example 2 - Standard Catalog Report with OPTIONS
4. Example 3 - VSAM Volume Summary
5. Example 4 - VVDS Report
6. Example 5 - ICF Volume Summary
7. Example 6 - Standard VTOC Report
8. Example 7 - Standard Label Report

---

Introduction

CBLVCAT has a standard report format which contains the most frequently required file information. Some columns of this report can display different fields depending upon the installation defaults and/or the run time options chosen. OPTION parameters can also be used to control the page geometry.

Example 1. Standard Catalog Report

A more readable replacement for an IDCAMS LISTCAT report (for ICF or VSAM catalogs), which is produced in a fraction of the time. The important data-component information is contained on one line only. Access to additional Catalog information is achieved either via the customised REPORT feature, or by using OPTION to alter the standard report content. Both these methods are shown in later examples.

e.g. List the Catalog identified by the DDNAME CBLV91.

   LC    DD=CBLV91
** Expiry: 2010-07-28 **

Example 1. Standard Catalog Report

```
Example: Standard Catalog Report (CRL Ref: vmxvfull)

Notes

1. **SPANNED** is a non-standard message appearing in the **BUFSP/IXL** column, indicating that the file is defined with the SPANNED attribute.

2. **-IMB--** in the **KL,RKP/BLK/IMB** column, indicates that the file was defined with the IMBED attribute.

3. A **V** suffix on the **LMAX** value, indicates that the loaded records are of variable length.

4. **IXL=n** in the **BUFSP/IXL** column, indicates the number of index levels.

To enable easy monitoring of file condition, CBLVCAT also displays warning indications (*,+*) when various, user controlled, threshold values are exceeded. This allows corrective action to be taken before serious problems occur. The warnings in this example are as follows:

1. **PCNT** (**ALL**) All primary space is used and file additions will cause secondary allocations.

2. **PCNT** (**nn.nn**) The file full percentage threshold has been reached - default 85% (See CBLNAME option **CBLVCPCF**).

3. **CISIZE (+nnnnn+)** Excessively large index Cl size.

4. **SEC** No secondary allocation defined (Highlighted as the file cannot be expanded).

5. **BUFSP** (nnnnn) The defined Bufferspace is too small.

Example: Standard Catalog Report (CRL Ref: vmxvfull)
**Example 2. Standard Catalog Report with OPTION**

**OPTION** can be used to control, among other things, the page geometry and the content of the standard report. It is applicable to both **LISTVCAT** and **LISTVTOC** operations, with parameters which relate specifically to the two different commands, as well as common parameters.

**e.g. Using **OPTION **to display non-default report fields of the Standard Report.**

```
Option  Unused  Avrl  CI/CA  S/C  Defined  * Non-default options.
```

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Example Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBLV10</td>
<td>[args]</td>
<td></td>
</tr>
<tr>
<td>OPTION</td>
<td>Unused</td>
<td></td>
</tr>
<tr>
<td>Avrl CI/CA</td>
<td>S/C Defined</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Catalog Report with OPTION</td>
<td></td>
</tr>
</tbody>
</table>

**Notes**

1. The fields in this report which are non-default, are as follows:

   1. **UNUSED**
      Display the percentage of allocated space which is unused (**UNUSED**), instead of the percentage which is used (**PCNT**).

   2. **AVRL**
      Display the average record length (**AVRL**), instead of the maximum record length (**LMAX**).

   3. **CI/CA**
      Display control intervals per control area (**CI/CA**), instead of Channel Programs executed (**EXCPS**).

   4. **S/C**
      Display Share Options and Space Class (**S/C**), instead of Share Options (**SHR**).

      **Note:** From release 9.80 onwards, column **S/C** or **SHR** is no longer displayed as part of the standard catalog report unless **CBLVCWS=’X’01’** bit is on. If this is the case, 2-digit years are displayed in the **TIMESTAMP/DEFINED** columns.

   5. **DEFINED**
      Display the date the file was defined (**DEFINED**), instead of the date the file was last closed by an operation that may have changed its contents (**TIMESTAMP**).

2. Any, or all, of these options may be made the installation default by setting the relevant switches in the **CBLNAME module**.

3. For some of the fields in this example, a ‘+’ (plus) sign is displayed after the number of records (**NRECS** - **nnnn**). This indicates that the record count has not been maintained by **VSAM** and the displayed value is **CBLVCAT’**s estimate.

4. Files shown as **TEMP** are temporary reusable files (**VSE** only), which means that the file has been defined as a work file. These files are not allocated until they are opened for output, at which time the required space is taken from unused remaining space. The total blocks/tracks these files will occupy is reported separately at the end of the report (in this example **NOALLOC=10**).
It is then possible to see if there is enough space left for all the files to be open at once (It is however, extremely unlikely that this would be required).

Example 3. VSAM Volume Summary

The volume summary follows the files, as shown in the previous examples. It appears on the standard Catalog report and the standard Catalog report with options, but is not user configurable. You can, however, report on volumes only, via the SUMMARY parameter:

e.g.

LC DD=CBLV05 SUMMARY

Example: VSAM Volume Summary (CBL Ref: vnxvsum)

Notes

1. When summary is used to suppress the report detail, the self-defining Catalog entry is still reported.

2. On the above report, this self defining entry has an asterisk in the SEC column. This indicates that the number of secondary extents has reached the warning threshold value (default 1 - see CBLNAME option CBLVCEXT). The actual number of extents is displayed after the asterisk. If the number of secondary extents multiplied by the secondary allocation plus the prime allocation is greater than the total allocation, this indicates that the primary allocation is split over multiple extents.

3. The Catalog to which the Volume Summary applies, owns dataspaces on three volumes. The percentage of space used on volume CBLV25 is flagged and a WARN 012 message is displayed, indicating that this percentage value has exceeded the user defined warning threshold (default 85% - see CBLNAME option CBLVCPGV).

4. Return code 06 indicates that a warning message was produced on this run (Additionally, every time CBLVCAT produces a return code, a WARN 001 message is produced).

Example 4. VVDS Report

Reporting directly on the contents of a VVDS is also available (See also Example 11 in the Customised report section for further use of this feature).

e.g.

LC VVDS=CBL104
Example: ICF Volume Summary (CBL Ref: vbxv5f8)

**Notes**

1. **IMB+REP** in the **BUFSP/IXL** column, indicates that the file was defined with the **IMBED** and **REPLICATE** attributes.

2. For SMS managed volumes, **NONVSAM** data sets are reported with the limited information contained in the VVDS.

3. Warning indications on this report not described in previous examples are as follows:

   1. **PCNT** (**FULL**)
      The file space is full and no secondary allocations have been specified.

   2. **FRSP CI CA**
      Unusable **free space** has been defined.

   3. **TIMESTMP** (**nn CI SPLIT**)
      The number of CI splits has reached the warning threshold (default 1 - see CBLNAME option CBLVCSCI).

Example 5. ICF Volume Summary

The VSAM concept of sub-allocated space does not exist for MVS ICF catalogs, therefore the volume summary for these catalogs contains fewer fields.

e.g.

<table>
<thead>
<tr>
<th>VVDS</th>
<th>CBLI04</th>
<th>TYPE</th>
<th>NRECS</th>
<th>PCNT</th>
<th>---- ALLOC TRACKS ----</th>
<th>FRSP</th>
<th>LMAX</th>
<th>KL,KRP</th>
<th>CISIZE</th>
<th>BUFSP</th>
<th>CI/CA</th>
<th>TIMESTMP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYSL.VVDS.VCLII04</td>
<td>ESDS</td>
<td>30** ** ALL**</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4089</td>
<td>4096</td>
<td>0</td>
<td>10</td>
<td>2007/08/05 17.34.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CICS.IVSTCICS.CDSD</td>
<td>KSDS</td>
<td>414</td>
<td>10.3</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>200V</td>
<td>22,0</td>
<td>22528</td>
<td>45568</td>
<td>18</td>
<td>2007/06/12 16.10.32</td>
</tr>
<tr>
<td>CICS.IVSTCICS.CDSD</td>
<td>KSDS</td>
<td>1</td>
<td>2.2</td>
<td>1</td>
<td>1</td>
<td>0*</td>
<td>505</td>
<td>512</td>
<td>46</td>
<td>2007/08/05 17.34.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CICS.IVSTCICS.CDSD</td>
<td>KSDS</td>
<td>300** ** ALL**</td>
<td>2</td>
<td>2</td>
<td>C=0*</td>
<td>4089</td>
<td>4096</td>
<td>8192</td>
<td>150</td>
<td>2007/08/05 17.34.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CICS.IVSTCICS.CDSD</td>
<td>KSDS</td>
<td>45</td>
<td>8.1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>8D</td>
<td>6,1</td>
<td>22528</td>
<td>45568</td>
<td>2</td>
<td>2007/08/05 17.34.00</td>
</tr>
<tr>
<td>CICS.IVSTCICS.CDSD</td>
<td>KSDS</td>
<td>4</td>
<td><strong>ALL</strong></td>
<td>1</td>
<td>1</td>
<td>0*</td>
<td>505</td>
<td>512</td>
<td>46</td>
<td>2007/08/05 17.34.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CICS.IVSTCICS.DFPhoton</td>
<td>ESDS</td>
<td>100** <strong>FULL</strong></td>
<td>10</td>
<td>10</td>
<td>0*</td>
<td>4089</td>
<td>4096</td>
<td>8192</td>
<td>100</td>
<td>2007/08/05 17.34.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICF.CAT.CBLI04</td>
<td>KSDS</td>
<td>19<strong>95.0</strong></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2041</td>
<td>2048</td>
<td>18</td>
<td>2007/10/31 16.04.23</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TMN.TESTMNAS.TMGT01</td>
<td>KSDS</td>
<td>53</td>
<td>50.9</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>8185</td>
<td>25,0</td>
<td>8192</td>
<td>17408</td>
<td>70</td>
<td>2007/08/05 17.34.00</td>
</tr>
<tr>
<td>TMN.TESTMNAS.TMGT01</td>
<td>KSDS</td>
<td>4</td>
<td><strong>ALL</strong></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1017</td>
<td>1024</td>
<td>15</td>
<td>2007/08/05 17.34.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICF.CAT.CBLI04</td>
<td>KSDS</td>
<td>47+</td>
<td>50.0</td>
<td>14</td>
<td>14</td>
<td>7</td>
<td>10</td>
<td>32400V</td>
<td>45,9</td>
<td>1024</td>
<td>186</td>
<td>2007/08/05 17.34.00</td>
</tr>
<tr>
<td>CICS.IVSTCICS.TMON</td>
<td>KSDS</td>
<td>150</td>
<td><strong>ALL</strong></td>
<td>1</td>
<td>1</td>
<td>0*</td>
<td>4089</td>
<td>4096</td>
<td>8192</td>
<td>150</td>
<td>2007/08/05 17.34.00</td>
<td></td>
</tr>
<tr>
<td>CICS.IVSTCICS.TMON</td>
<td>KSDS</td>
<td>27</td>
<td>5.4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2000</td>
<td>10,0</td>
<td>2048</td>
<td>6656</td>
<td>252</td>
<td>2007/08/05 17.34.00</td>
</tr>
<tr>
<td>TMON.TESTMNAS.TMGT01</td>
<td>KSDS</td>
<td>281</td>
<td>26.8</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>4096</td>
<td>4096</td>
<td>8192</td>
<td>150</td>
<td>2007/08/05 17.34.00</td>
<td></td>
</tr>
</tbody>
</table>

Example: ICF Volume Summary (CBL Ref: vbxv5f8)

A more readable replacement for IEHIST/ LVTOC. The important VTOC information is contained on only one line. Other VTOC information is available either through OPTIONS to the standard report, or via the customised REPORT feature (see Customised Reports).

e.g. List the VTOC of the volume CBLT93

```
OPTION FREETAB * Display freespace in a table at the end of the report.
```

```
LV VOL=CBLT93
```

```
** VMXTFULL CTL M *** LEVEL 001 +++ 97/06/23 16:11:51 *id *
```

```
CBLVCAT REL 2.12 AT COMPUTE (Bridgend) Ltd - Wales (OS) VM/CMS=VMNB3 13.35 FRI 26 JUN 2009 PAGE 1
```

1. This report is for an FBA device. For a CKD device, the CISIZE field would be replaced by the CYL/HD field. The CBLVCAT messages shown in the above example are as follows:

- **EXPIRES** ("EXPD")
  Expiry date is less than or equal to todays date (if option EXPD is in force).

- **EXPIRES** (PERMANENT)
  An expiry date of 1999/366 or, if defined as an absolute Julian date and not implied via a retention period in days, 1999/365 (if option PERM is in force).

- **INFO** (OWNED BY VSAM CATALOG)
  This volume is a prime, or candidate, volume of a VSAM catalog. It is therefore not available to other VSAM catalogs.

- **INFO** (NOT LAST VOL)
  Indicates that a sequential file has more extents on another volume, or that a file has not been closed.

- **INFO** (FILE SERIAL CBLT13)
  The file serial number is different from the disk volume serial number (Possibly caused by changing the VOLID after creation of the file).

- **INFO** (**INVALID EXTENT**)
  The extent is not totally within the extent limits in the VTOC Format 4 label.
### Example 7. Standard Label Report

For VSE only, a more readable and concise replacement for the **Lserv** label information area listing.

**e.g.**

```
LISTLABL
```

---

<table>
<thead>
<tr>
<th>PARTN FNAME</th>
<th>STD LABEL DSN</th>
<th>SYSDO VOLUME CAT/TYP</th>
<th>DISP</th>
<th>EXPIRES</th>
<th>BUFNI</th>
<th>BUFND</th>
<th>START/PRECS</th>
<th>ALLOC/SRECS</th>
<th>RECCS</th>
<th>INFO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (V)</td>
<td>N,K</td>
<td>99</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 (S)</td>
<td>2010/02</td>
<td>5510</td>
<td>7851</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td>32767</td>
<td>32767</td>
<td>EXT=000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS02 SYSHR2</td>
<td>7</td>
<td>32768</td>
<td>32768</td>
<td>EXT=001</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS03 SYSHR3</td>
<td></td>
<td>65535</td>
<td>77566</td>
<td>EXT=002</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PARTN FNAME</th>
<th>STD LABEL DSN</th>
<th>SYSDO VOLUME CAT/TYP</th>
<th>DISP</th>
<th>EXPIRES</th>
<th>BUFNI</th>
<th>BUFND</th>
<th>START/PRECS</th>
<th>ALLOC/SRECS</th>
<th>RECCS</th>
<th>INFO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>PARTN FNAME</th>
<th>STD LABEL DSN</th>
<th>SYSDO VOLUME CAT/TYP</th>
<th>DISP</th>
<th>EXPIRES</th>
<th>BUFNI</th>
<th>BUFND</th>
<th>START/PRECS</th>
<th>ALLOC/SRECS</th>
<th>RECCS</th>
<th>INFO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SYSS01 SYSHR1 (S)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

### Notes

1. No parameters are supplied to this command, which must be complete on one logical record. The order of the report is:
   1. **Partition** standard labels (temporary and permanent).
   2. **Class** (Dynamic partition) standard labels.
   3. **System** standard labels.

2. If any group of labels is not accessible to CBLVCAT (possibly because it is currently being updated), a message indicating the group which could not be accessed is included in the report (No warning is given if a particular group of labels is empty).

```
F4
*** Inaccessible ***
```
In the PARTN column, a system standard label is indicated by STD and a permanent or dynamic partition label is indicated using the partition-id. If the partition label is temporary, (T) is appended to the partition-id.

For non-VSAM file labels that have names not beginning IJSYS, the warning "NO" is displayed in the SYSNO column if no logical unit exists within the label definition. Similarly "NO" is displayed in the VOLUME column of any non-VSAM label if no EXTENT information is defined.

In the CAT/TYP column, VSAM files will show the name of the Catalog that owns the file if CAT=xxxx was specified on the DLBL, or (V) otherwise. Sequential files will show (S) and Direct Access files will show (DA).

In the DISP (file disposition) column, the abbreviations used are N (NEW), O (OLD), K (KEEP), D (DELETE) and T (DATE).

The START/PRECS column contains information which varies according to file type. For sequential files, direct access files and VSAM files (or data spaces) created with the UNIQUE option, it contains extent information (if specified). This is either the sequential track number (relative to zero) where the extent is to begin (CKD), or the physical block where the extent is to begin (FBA). For SAM files which reside in a VSAM space it contains the number of records for the primary allocation (if specified).

The ALLOC/SRECS column contains information which varies according to file type. For sequential files, direct access files and VSAM files (or data spaces) created with the UNIQUE option, it contains the extent allocation (if specified) in tracks (CKD) or blocks (FBA). For SAM files which reside in a VSAM space it contains the number of records for the secondary allocation (if specified).

As can be seen from the above example, if a DLBL has been defined with more than one extent (i.e. more than one EXTENT card), the additional extent information is included on the following line (the unchanged DLBL information is not repeated). In this case the INFO column contains the extent sequence number in the form EXT=nnn.

Customised Reports

1. Introduction
2. Example 8 - Basic Customised Report
3. Example 9 - Selective Customised Report
4. Example 10 - Advanced Customised Report
5. Example 11 - Customised VVDS Report
6. Example 12 - Combined VTOC Report
7. Example 13 - Combined Catalog Report
8. Example 14 - Combined Catalog and VTOC Report
9. Example 15 - Combined VTOC Report for all VTOCs
10. Example 16 - Combined Catalog Report for all Catalogs
11. Example 17 - Combined Report for all Catalogs & VTOCs
12. Example 18 - Free Space across all VTOCs

Introduction

If the standard Catalog or VTOC report does not provide the required format or content, it is possible to produce a customised report containing the necessary information, in a format tailored to your requirements. (Customised printing for LISTLABL is not supported). Customisation is accomplished using one or more of the following commands/parameters, together with their associated parameters/sub-parameters.

OPTIONS
Used to control the page geometry for the report and the level of data reported for each file. It can also be used to control the fields in the standard report.

SUBSET
Allows the user to select which files should be included in the report.

REPORT
Each parameter of the REPORT command determines which fields will appear in a customised report and the order in which they appear (See Summary of Syntax for a complete list of REPORT parameters, their default field widths and their column format and heading).

SORT
Lines of output produced by a REPORT command can be sorted into ascending or descending order using the SORT (synonym SORTA) or SORTD parameters.

STOPAFT
For use with the REPORT VCAT/VTOC SORT parameter. It allows the user to limit the number of files to be displayed in the report. STOPAFT is ignored if not used in conjunction with REPORT VCAT/VTOC SORT.

MERGE
Used to combine multiple reports into a single listing.
Example 8. Basic Customised Report

`REPORT` can be used, in conjunction with a `LISTVCAT / LISTVTOC` operation, to control the fields within the report and the order in which they are displayed.

e.g.

```
REPORT VCAT DSN SORTD NRECS TYPE
LC DD=CBLI94 NOINDEX
```

Example: Basic Customised Report (CBL Ref: vmxvsimp)

Notes

1. In this example the three fields of the report are:
   1. Dataset name (DSN).
   2. Number of records in the file (NRECS).
   3. File type (TYPE).

2. The `SORTD` parameter results in a report which is sorted, in descending order, on the fields which follow the parameter.

   The `NOINDEX` `LISTVCAT` option, causes KSDS and AIX index lines to be suppressed.

Example 9. Selective Customised Report

So far, the examples have displayed the full contents of a catalog or VTOC. Most of the time you will probably only be interested in a particular file or set of files.

e.g. Catalog report on a `SUBSET` of files.

```
REPORT VCAT DSN 30 SORTD NRECS 14
OPTIONS NOVOL NOASSOC NOINDEX
LC DD=CBLV03
SUBSET IGN=/CICS KEY=TEST NRECS=1 TYPE=EK
```

Example: Selective Customised Report (CBL Ref: vmxvsel)
Notes

1. The **REPORT** command is described in the previous example. However, any of its parameters can also be supplied with a numerical argument which overrides the implicit column width, e.g. **NRECS 14**

2. The **SUBSET** parameter is used for file selection. Only files obeying all the subsequent sub-parameters are selected. i.e. a logical **AND** (The exceptions are **KEY** and **IGN** which use a logical **OR**). The word **SUBSET** is optional as its associated sub-parameters are unique.

   - **NOVOL** suppresses Volume information.
   - **NOASSOC** suppresses file association information.
   - **NOINDEX** suppresses index lines for AIX and KSDS files.

3. The files which are included in this report must obey the following conditions:

   1. **NRECS=1**
      All files with 1, or more, records i.e. all non-empty files.

   2. **TYPE=EK**
      All ESDS (E) or KSDS (K) files.

   3. **KEY=TEST**
      All files with a Dataset name starting with TEST.

   4. **IGN=CICS**
      Ignore all files with a name containing the string ‘CICS’.

4. Remember that **NRECS** and **TYPE=xxx** will be processed as a logical **AND** (i.e. both conditions must be true). A file that satisfies this condition will then have to satisfy **IGN=CICS OR KEY=TEST**.

5. The customised report produced will also have been sorted in descending order (**SORTD**). The sort is performed on the fields which follow **SORTD** in the parameter list (In this case the number of records - **NRECS**).

Example 10. Advanced Customised Report

A customised report can also be an invaluable tool to more experienced users, who prefer to analyse information themselves. By allowing the report to focus on particular aspects of file activity, the user can produce a clear, concise report which makes for easier analysis.

e.g. Catalog report relating to I/O performance.

```
REPORT VCAT DSN 25 EXCP5 RECSTATS 28 IXL CISIZE BUFS P 12 NSEC
LC DD=CBLV93
   TYPE=K * Select:  KSDS only
   KEY=A.DEMO  * DSN's starting 'A.DEMO' only.
```

<table>
<thead>
<tr>
<th>USERCAT CBLV93 (3370)</th>
<th>EXCP5</th>
<th>-- RECORD STATISTICS --</th>
<th>IXL</th>
<th>CISIZE</th>
<th>BUFS P</th>
<th>NSEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>o A.DEMO.HIGH.EXCP5.INP</td>
<td>17.2K</td>
<td>9760</td>
<td>9779</td>
<td>2048</td>
<td>16000</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>6253</td>
<td></td>
<td>3</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.ORDINARY.1</td>
<td>159</td>
<td>3</td>
<td>2</td>
<td>197K</td>
<td>1536</td>
<td>4096</td>
</tr>
<tr>
<td></td>
<td>165</td>
<td></td>
<td>1</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.ORDINARY.2</td>
<td>60</td>
<td>2</td>
<td>39</td>
<td>1536</td>
<td>4096</td>
<td></td>
</tr>
<tr>
<td></td>
<td>46</td>
<td></td>
<td>1</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.ORDINARY.3</td>
<td>2075</td>
<td>9</td>
<td>1424</td>
<td>18.7K</td>
<td>1536</td>
<td>4096</td>
</tr>
<tr>
<td></td>
<td>649</td>
<td></td>
<td>1</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.ORDINARY.4</td>
<td>2674</td>
<td>1</td>
<td>4415</td>
<td>1</td>
<td>2048</td>
<td>16000</td>
</tr>
<tr>
<td></td>
<td>939</td>
<td></td>
<td>2</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.ORDINARY.5</td>
<td>1646</td>
<td>9</td>
<td>1047</td>
<td>8738</td>
<td>1536</td>
<td>4096</td>
</tr>
<tr>
<td></td>
<td>591</td>
<td></td>
<td>1</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.MORE.IX.EXCP5</td>
<td>2475K</td>
<td>18</td>
<td>448</td>
<td>563K</td>
<td>1806K</td>
<td>3072</td>
</tr>
<tr>
<td></td>
<td>6830K</td>
<td></td>
<td>3</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.XYZ.1</td>
<td>12K</td>
<td>21</td>
<td>3</td>
<td>10.9K</td>
<td>67.8K</td>
<td>1536</td>
</tr>
<tr>
<td></td>
<td>579</td>
<td></td>
<td>1</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.XYZ.2</td>
<td>225</td>
<td>5</td>
<td>2</td>
<td>287</td>
<td>1536</td>
<td>4096</td>
</tr>
<tr>
<td></td>
<td>231</td>
<td></td>
<td>1</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o A.DEMO.XYZ.3</td>
<td>440</td>
<td>9</td>
<td>7</td>
<td>64</td>
<td>2028</td>
<td>1536</td>
</tr>
<tr>
<td></td>
<td>187</td>
<td></td>
<td>1</td>
<td>512</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Example: Advanced Customised Report (CBL Ref: vmxvadv)
Example 11. Customised VVDS Report

There is not always a match between BCS and VVDS entries. This can be caused either by Catalogs being removed without the appropriate entry in the VVDS being deleted, or by a job failure. These orphaned entries increase system overheads and should be deleted as part of a manual housekeeping operation (Such housekeeping, of course, should be conducted by experienced personnel only).

Customised reporting on the VVDS (listing the Catalog and Component) helps the user to detect these orphaned entries (provided the removed catalogs are known). The IDCAMS command DELETE VVR can then be used to delete the relevant VVR (VSAM Volume Record) entries from the VVDS and the VTOC.

e.g. Customised report containing Catalog and component information only and also sorted primarily by Catalog and then by Component.

```
REPORT VCAT SORT CATALOG 30 COMPONENT 30
LISTVCAT VVDS=CB9043
```

Example: Customised VVDS Report (CBL Ref: vbxv56)
Notes

1. As the report is sorted by Catalog, all entries relating to a non-existent Catalog will appear consecutively on the report making location easier.

2. If the Catalog SYS1.ICFCAT.OLDCAT no longer exists, the VVR records in VVDS CB9043 which reference this Catalog are therefore redundant.

Example 12. Combined VTOC Report

The MERGE parameter may be used on a LISTVTOC or LISTVTOC command, provided a REPORT statement for the appropriate operation has already been supplied. It causes the output from the current command, to be merged with the output from the command that follows. Thus, the user can merge the listings from two or more LISTVTOC / LISTVTOC commands and produce a single customised report.

e.g. Combining three VTOC Reports and sorting them in descending allocation size.

Example: Combined VTOC Report (CBL Ref: vmxmerl)

Notes

1. Using MERGE results in DATASET NAME being used as the heading for the DSN column.

2. If the applicable REPORT statement does not include a SORT parameter, a default sort (ascending) is performed using the first entry in the REPORT list as the primary sort field.

Example 13. Combined Catalog Report

A MERGE of several operations which reference the same catalog or VTOC, but consist of different SUBSET values, effectively produces a logical OR condition for the subsets. However, any entry which would be selected by more than one SUBSET value will not be reported twice.

e.g. Using MERGE to select files with a prescribed number of CI/CA splits OR secondary extents.
Example 14. Combined Catalog and VTOC Report

**MERGE** can also be used to produce a report which combines Catalog and VTOC reports. For this type of report, the headings and sort positions of the report are taken from the **REPORT** parameters applicable to the last operation of the merged output. This means that, when combining **LISTVCAT** and **LISTVTOC** commands, it is possible for a **LISTVCAT** heading line, to be displayed over **LISTVTOC** data and vice-versa.

e.g. Combined report in creation/defined date order (Although these fields are not identical, they both show the age of the file).
1. A left margin width of 2 characters is established in both REPORT statements in order to indent the report column entries.

2. The **DEFINED** and **CREATED** fields have slightly different formats and widths: ccyy/mm/dd* (11 characters) and ccyy/mm/dd (10 characters) respectively. To ensure that the columns that follow line up correctly, both columns are defined with a width of 15. Note that, if a column width is specified then an intervening blank between the column data is no longer inserted automatically.

3. Similarly, the VTOC **TYPE** field and VCAT **TYPE** field have different widths (7 characters and 8 characters respectively). Therefore, to maintain alignment of the DSN columns that follow, both TYPE columns are defined with a width of 9.

4. In this example, the **SORT** position and headings from the REPORT VTOC operation are not used (they are taken from the REPORT VCAT as LISTVCAT is the last command). Although not illustrated in this example, any REPORT fields which do not apply to the LISTVTOC operation are left blank for LISTVTOC report lines.

**Example 15. Combined VTOC Report for all VTOCs**

Using **MERGE** in conjunction with **SORT** and **DEV=ALL**, creates a VTOC report over all volumes.

e.g. VTOC report over all Volumes, sorted by Blocksize and containing columns for Data Set Name, Volume Serial Number and Blocksize.

```
<table>
<thead>
<tr>
<th>DATASET NAME</th>
<th>VOLUME</th>
<th>BLKSIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SASBR.RW.SASDATA</td>
<td>CBLT06</td>
<td>32760</td>
</tr>
<tr>
<td>TEST.BASE650.ONLINE_LOADLIB</td>
<td>CBLT06</td>
<td>32760</td>
</tr>
<tr>
<td>TEST.SOS111.LINKLIB</td>
<td>CBLT06</td>
<td>32760</td>
</tr>
<tr>
<td>TEST.AKS110.LINKLIB</td>
<td>CBLT06</td>
<td>32760</td>
</tr>
<tr>
<td>TEST.SOS111.ACLIB</td>
<td>CBLT06</td>
<td>32760</td>
</tr>
<tr>
<td>SASBR.ETS.SASDATA</td>
<td>CBLT06</td>
<td>32760</td>
</tr>
<tr>
<td>DSN120.DSNLOAD</td>
<td>CBLT07</td>
<td>23200</td>
</tr>
<tr>
<td>IMS130A.RLRESLIB</td>
<td>CBLT07</td>
<td>23200</td>
</tr>
</tbody>
</table>
```

Example: Combined VTOC Report for all VTOCs (CBL Ref: vmxmermt)

**Notes**

1. See **DEV=ALL** for operating system conditions.

**Example 16. Combined Catalog Report for all Catalogs**

Using **MERGE** in conjunction with **SORT** and **REF=ALL**, creates a Catalog report over all Catalogs.

e.g. Catalog report over all Catalogs, containing Number of records, Data Set Name and Catalog Name. Index entries are suppressed (**NOINDEX**) and a sort is carried out in descending order (**SORTD**) of **NRECS**.

```
```

2013-10-29 15:43:01  CBLVCAT Manual  24
Example 17. Combined Report for all Catalogs and VTOCs

A combined Catalog/VTOC report over all Catalogs/Volumes can also be combined with any other report. To accomplish this a second MERGE parameter is needed, as the first MERGE is used to produce the original combined Catalog/VTOC report.

e.g. Using MERGE in conjunction with SORT, DEV=ALL MERGE and REF=ALL MERGE, to obtain a single report over the whole of the system (VTOC output, of course, will have no Catalog name).

Example: Combined Report for all Catalogs and VTOCs (CBL Ref: vmxmermb)

Notes

1. This example is only applicable to ICF Catalogs and requires DFP 3.1 or higher.

2. The SORT parameter in this example could have been omitted, as the default is an ascending sort using the first entry in the REPORT list as the primary sort field (in this case DSN).
Example 18. Free Space across all VTOCs

The **SORT** feature, used in conjunction with **DEV=ALL MERGE, SUMMARY** and **FREETAB**, enables sorting of freespace over all volumes. It therefore allows the user to create a system **freespace** map.

```
| REPORT VTOC   VOLUME SORTD ALLOC START  |
|-------------|-------------|-------------|-------------|
| LISTVTOC    DEV=ALL MERGE              |
| SUMMARY FREETAB     * Freespace over all volumes |
| STOPAFT 100         * 100 largest freespace areas |
| VOLUME   ALLOC    START                |
| ------   -----    -----                |
| CBLT03  154786  2                        |
| CBLT03  139536  192288                    |
| CBLT03  118064  213760                    |
| CBLT01  104098  557542                    |
| CBLT01  71258   563500                    |
| CBLT01  20544   691968                    |
| CBLT01  15650   47138                      |
| CBLT03  13792   683336                    |
| CBLT01  12725   634758                    |
```

Example: Free Space across all VTOCs (CBL Ref: vmxtall)

**Notes**

1. See **DEV=ALL**, for operating system conditions.

2. The storage and run-time requirements could be considerable. Although CBLVCAT is only producing a relatively small number of print lines (governed by the free extents available and the user specified **STOPAFT**), all files in all volumes will be processed.

## IDCAMS DEFINE and Reorganisation

1. Introduction
   2. Supported File Types
   3. Output File
   4. Selection
   5. Security Keywords
   6. Warning Messages
   7. IX Allocation
   8. Installation Standards
   9. Example 19 - DEFINE without TUNE
   10. Example 20 - DEFINE with TUNE
   11. Example 21 - Skeleton Reorganisation Jobstreams

### Introduction

The **DEFINE** (synonym **DEF**) parameter, requests CBLVCAT to produce an additional output file of IDCAMS **DELETE/DEFINE** parameters. This enables the modelling of a file definition based on an existing file of similar characteristics. It also enables the retrieval of up to date definitions directly from the catalog, thus eliminating the possibility of re-introducing back level attributes.

The output variations available are:

1. IDCAMS **DEFINE** attributes without file tuning recommendations.
2. IDCAMS **DEFINE** attributes with file tuning recommendations.

See examples 19, 20 and 21 later in this section to illustrate these variations.
**Supported File Types**

ESDS, KSDS, RRDS, AIX, PATH and BLDINDEX definitions. Other types, such as SAM/ESDS, NONVSAM, USRCAT etc. are not currently supported.

**Output File**

**VSE**
The output is to the system unit SYSPCH. In addition to being a real punch unit (unlikely these days), this can be maintained on the **POWER PUN** queue, or directed to a disk via the system file IJSYPH. VM/VSE users can route the punch output to a virtual machine's reader.

**MVS**
The output is to **SYSPUNCH**, with default attributes of RECFM=FB LRECL=80 BLKSIZE=800 when assigned to disk. If SYSPUNCH is directed to a member of a PDS, simultaneous update is protected by means of the ENQ/DEQ facility. To ensure compatibility with the **SPF EDITOR**, the primary 8-byte name used for ENQ is 'SPFEDIT', while the secondary name used is the Data Set Name itself.

**CMS**
The output is to the virtual machine's punch. CBLVCAT will assume that VSE VSAM-type definitions are required. Therefore, users running against MVS VSAM (non-ICF) catalogs, will need to set the CBLVCSW4 X'20' bit on, either at run-time using **OPTION**, or as a default in the **CBLNAME** module, to produce the correct output.

**Selection**

The standard **SUBSET** commands (e.g. **KEY=ABC**), should be used to control which datasets are required in the **DEFINE** output.

**Security Keywords**

**PASSWORD** related items, such as ATMPTTE, AUTHORISATION, CODE, CONTROLPW, MASTERPW, READPW, etc. are ignored.

**Warning Messages**

When using **DEFINE** in conjunction with **TUNE** it is possible for warning messages to appear in the **DEFINE** parameters (One reason would be if CBLVCAT were recommending a large change to the defined average record length). These warnings indicate that local knowledge may be required (see Guide to VSAM Tuning for further details).

A warning message is always given when a skeleton reorganisation deck is produced as it will require user attention.

**Removal of these messages**, to ensure successful IDCAMS processing, is taken as acceptance of responsibility for the job by the user (See the descriptions of **WARN 014 - WARN 019** in the Messages chapter).

**IX Allocation**

CBLVCAT does not produce an index allocation in the **DEFINE** parameters. Specifying only a **DATA** allocation forces IDCAMS to calculate the **INDEX** allocation. This allocation, which is in addition to the data allocation, is based upon the control interval size, record size, device type and other data set attributes. Allowing the **INDEX** allocation to default, ensures that other changes made in the **DEFINE** parameters, prior to definition, are automatically reflected in the index. This ensures that the INDEX allocation will never get 'out of step'.

**Installation Standards**

The production of **DELETE/DEFINE** parameters is, primarily, governed by a switch setting in **CBLNAME** (bit CBLVCSW6 X'80').

<table>
<thead>
<tr>
<th>Switch</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ON</strong></td>
<td>Results in option <strong>DEFINE</strong> as the default.</td>
</tr>
<tr>
<td><strong>OFF</strong></td>
<td>Results in option <strong>NODEFIN</strong> as the default.</td>
</tr>
</tbody>
</table>

Output from the **DEFINE** parameter is controlled by the **CBLNAME** switches CBLVCSW6, CBLVCSW7 and CBLVCSW8. See **CBLNAME** (for details on all switches). It can also be conditioned at run time, using CBLVCSW6, CBLVCSW7 and CBLVCSW8 as run-time options. The switch settings which affect **DELETE/DEFINE** and reorganisation parameters are as follows:
Catalog Name

The **DEFINE** parameters can include the Catalog name statement. If excluded the define defaults to the 'Order of Catalog Selection'.

The **CBLVCSW6 X’10’** bit in **CBLNAME** controls this option.

| ON  | Suppress Catalog name |

The **DELETE** parameters should include the Catalog name for safety reasons. However it can be suppressed and, if excluded, defaults to the 'Order of Catalog Selection'. **This should be used with caution.**

The **CBLVCSW8 X’10’** bit in **CBLNAME** controls this option.

| ON  | Suppress Catalog name |

Reorganisation

The reorganisation facility produces a skeleton jobstream which **requires user attention**. A suitable warning message, **WARN 019**, is included in the jobstream before the IDCAMS **DELETE** to emphasise this point. It is assumed that the **Compute (Bridgend) Ltd** product **SELCOPY** is to be used to perform the file back up/restore operation.

The **CBLVCSW6 X’08’** bit in **CBLNAME** controls this option.

| ON  | Create Reorganisation deck |

Backup

**Reorganisation** defaults to tape as the backup medium.

The **CBLVCSW6 X’04’** bit in **CBLNAME** controls this option.

| ON  | Use disk for Reorganisation work file |
| OFF | Use tape for Reorganisation work file |

**PATH**

Path decks, selected by the specified **SUBSET** parameter(s), can be filtered out.

The **CBLVCSW7 X’80’** bit in **CBLNAME** controls this option.

| ON  | PATH decks suppressed |

**BLDINDEX**

A **BLDINDEX** deck is used to construct the data in the Alternate Index dataset. The Prime and Alternate keys from the Base Cluster are used to compose the data portion of the AIX. The **BLDINDEX** deck, if requested, follows the **DEFINE AIX** deck.

The **CBLVCSW7 X’40’** bit in **CBLNAME** controls this option.

| ON  | **BLDINDEX** deck(s) suppressed |

Comments

The old define values are included in the define parameters in a 'commented out' form (**/* in positions 2 and 3). This can be suppressed.

The **CBLVCSW7 X’20’** bit in **CBLNAME** controls this option.

| ON  | Suppress **DEFINE** comments |

Notes

The tuning notes available in the SYSLST/SYSLIST report can also be part of the SYSPCH/SYSPUNCH output.

The **CBLVCSW7 X’10’** bit in **CBLNAME** controls this option.

| ON  | Include Notes |

**CLUSTER**

Cluster parameters selected by the specified **SUBSET** parameter(s) can be filtered out.

The **CBLVCSW7 X’08’** bit in **CBLNAME** controls this option.

| ON  | Cluster deck(s) suppressed |

**AIX**

AIX parameters selected by the specified **SUBSET** parameter(s) can be filtered out.

The **CBLVCSW7 X’04’** bit in **CBLNAME** controls this option.

| ON  | Aix deck(s) suppressed |

**DELETE**

The **DELETE** parameters preceeding the requested **DEFINE** parameters can be commented out (**/* in positions 2 and 3).

The **CBLVCSW7 X’02’** bit in **CBLNAME** controls this option.

| ON  | **DELETE** deck(s) not commented out |

**JCL**

**MVS / VSE** job control may be wrapped around the **DEFINE** parameters.

The **CBLVCSW7 X’01’** bit in **CBLNAME** controls this option.
CMS with MVS output is discussed under the heading Output File.

Example 19. DEFINE without TUNE

It is possible to reproduce the IDCAMS DEFINE parameters for a file/subset of files as they are currently defined, with no tuning recommendations.

E.g.

LC DD=CBLV04 KEY=TSTX.ASSOC DEFINE CBLVCSW7=X'01' * Suppress JCL

Reproduction of the typical output to SYSPCH/SYSPUNCH as follows:

```csh
/* DEL TSTX.ASSOC.TRANS.XREF */
/* CLUSTER */
/* PURGE */
/* CATALOG (VSAM.R1.TEST.VOL04) */

DEF CLUSTER (NAME (TSTX.ASSOC.TRANS.XREF)) INDEXED /* KSDS */
  INDEXED
  RECORDSIZE ( 50, 50)
  SPEED
  TO ( 99366)
  NONSPANNED
  FREESPACE ( 30, 10)
  KEYS ( 22, 0)
  IMBED
  NOREPLICATE
  SHAREOPTIONS (2,3)
  NOERASE
  UNORDERED
  NOREUSE
  USECLASS (0 P)
  SUBALLOCATION
  DATA (NAME (TSTX.ASSOC.TRANS.XREF.DATA))
    VOLUMES ('CBLV04'
               'CBLV32'
               )
    CISZ ( 3072)
  CYLINDERS ( 120, 10)
  INDEX (NAME (TSTX.ASSOC.TRANS.XREF.INDEX))
    VOLUMES ('CBLV04'
               'CBLV32'
               )
    CISZ ( 2560)
)
CATALOG (VSAM.R1.TEST.VOL04)
```

Example: IDCAMS DEFINE without TUNE (CBL Ref: vmxvdefo)

Notes

1. In this example the three fields of the report are:

2. This example reproduces the DELETE/DEFINE parameters for all datasets with DSN starting with TSTX.ASSOC, from the catalog referenced by DDNAME CBLV04

3. JCL is suppressed using a run-time override option (CBLVCSW7=X'01').

Example 20. DEFINE with TUNE

If the DEFINE and TUNE parameters are both specified, CBLVCAT’s tuning recommendations are included in the IDCAMS DELETE/DEFINE parameters.

E.g.

LC DD=CBLI03 DEFINE TUNE KEY=/>.FEMP IGN=/AIX IGN=/PATH

Reproduction of the typical output to SYSPCH/SYSPUNCH as follows:
Example 20. DEFINE with TUNE

**Notes**

1. This example will reproduce the DELETE/DEFINE parameters and include tuning recommendations for datasets (referenced by Catalog CBLI03) with a DSN containing the string .FEMP. It will also exclude dataset names containing the strings AIX or PATH.

2. Ignoring AIX and PATH entries can also be set as an installation default using the CBLVCSW7 X'84' switches, as described earlier in this section, under the heading 'Installation Standards'.

3. Commented attributes show the original being replaced. These can be suppressed with the CBLVCSW7 X'20' switch.

4. Comments on the DEL statement can also be suppressed with the CBLVCSW7 X'02' switch.

Example 21. Skeleton Reorganisation Jobstreams

The CBLVCSW8 switch, together with the LISTVCAT command and DEFINE parameter, produces skeleton reorganisation jobstreams. The jobstreams use the Compute (Bridgend) Ltd product SELCÔPY is to be used to perform the file back up/restore operation.

**Suggested use**

A skeleton Reorganisation jobstream would be a useful building block, to users such as Operation Support, when liaising with on-call Technical Support colleagues. It presents the user with the fundamentals of a jobstream, for discussion and subsequent modification to in-house standards.

**Caution**

This is not intended as a back-up facility, only as a guide to a quick reorganisation. The jobstreams are provided to assist the user, but will require attention to comply with installation standards etc. WARN 019, is included in the jobstreams before the IDCAMS DELETE to emphasise this point. Removal of this warning message, is then taken as acceptance of responsibility for the job by the user, who should ensure that adequate back-up exists for the file(s) being reorganised.
The **CBLVCSW6** x'04' switch selects the medium for the intermediate file, which is needed to perform the back-up/restore (See the description in **Installation Standards**).

### VSE Skeleton

A reorganisation jobstream with **tape** as the intermediate file. e.g.

```
LC  DD=IJSYSUC  KEY=EXR.MASTER  DEFINE
CBLVCSW6 X'08'  * Reorganise(X'08') Tape(X'00')
```

Reproduction of the typical output to **SYSPCH** as follows:

```
// JOB CBLDEF01
// DLBL IJSYSUC,'FBA.USER.CATALOG.A',,VSAM
// ASSGN SYS001,TAPE
// PAUSE PLEASE LOAD TAPE TO SYS001 FOR CBL REORG
// TLBL TAPE01,'CBL.REORG.TEMP',001
// DLBL REORGDS,'EXR.MASTER',,VSAM
// EXEC SELCOPY,SIZE=AUTO
READ REORGDS KSDS
WRITE TAPE01  RECFM=VB B=32760
/*
// EXEC IDCAMS,SIZE=AUTO
*** WARN  016 *** LARGE ALLOC CHANGE
*** WARN  019 *** SKELETON DECK ONLY - ATTENTION REQUIRED
DEL       EXR.MASTER
CLUSTER
PURGE
CATALOG (FBA.USER.CATALOG.A)
(\________________________________________________________________________
\________________________________________________________________________
)(The generated DEFINE control cards for IDCAMS)
```

Example: VSE Skeleton (CBL Ref: vmxrorg1/2)

**Notes**

This example produces, in addition to the Catalog report, a Reorganisation Jobstream and **DEFINE** parameters for the file whose name commence with the string 'EXR.MASTER'.

See **CBLNAME** for a description of the **CBLVCSW6** switch settings.

### MVS Skeleton

A reorganisation jobstream with **disk** as the intermediate file. e.g.

```
LC  DD=ICFCATC  KEY=CICS161  DEFINE
CBLVCSW6 X'C'0C'  * Reorganise(X'C08') Disk(X'C04')
```

Reproduction of the typical output to **SYSPUNCH** as follows:
Example: MVS Skeleton (CBL Ref: vmxrorg3/4)

Notes
This example produces, in addition to the Catalog report, a Reorganisation Jobstream and DEFINE parameters for the file whose name commence with the string 'CICS161'.

See CBLNAME for a description of the CBLVCSW6 switch settings.
Guide to VSAM Tuning

This chapter illustrates and explains the output from CBLVCAT if the TUNE parameter is specified. The only exception is TUNE when used to produce a tuned DELETE/DEFINE deck, which is covered in the section IDCAMS DEFINE and Reorganisation of Guide to List Output.

Chapter Summary of Syntax contains a full list of all functions and their associated parameters and sub-parameters.

Detailed descriptions of functions, parameters and sub-parameters in alphabetical order, are available in chapter A-Z Reference.

Tuning Considerations

1. File Selection
2. Environment Selection
3. On-line and Batch Files
4. Alternate Index Files
5. Database Files
6. Reusable Files
7. Average Record Length Estimation
8. Growth and Freespace

Also see section Tuning Principles for CBLVCAT’s approach to tuning.

The user is able to influence CBLVCAT’s tuning routines using TUNE sub-parameters (See Summary of Syntax for a full list). These sub-parameters can either be used to keep the tuning recommendations within prescribed limits (e.g. restricting the CISIZE) or to override specific file attributes and have the tuning recommendations based on these overrides (e.g. the number of records in the file - RECORDS). The following include specific instances when some of these parameters would be used:

File Selection

Tuning is usually carried out on a particular file/set of files. It is possible to select the files you want to tune using the SUBSET parameter and its associated sub-parameters.

Environment Selection

Tuning is usually carried out on a particular file/set of files. It is possible to For optimal tuning, the operating system and the type of DASD have to be taken into account (The operating system affects the range of physical record sizes available to VSAM, while the disk architecture will determine which CI sizes are most efficient). The following TUNE sub-parameters relate to environment selection:

1. TUNE (sys)
   TUNE specifies that tuned output is required and, optionally, that the tuning is for a particular operating system. If TUNE only is supplied, tuning will be based on the current operating system. However, if an argument is supplied, tuning will be based on that argument (This facility is useful when migrating to a different operating system, as the files to be transferred can be tuned prior to the migration). The arguments allowed are: MVS and VSE.

2. DEV=nnnn
   DEV=nnn allows you to tune files for device types which differ from those currently in use (e.g. migrating from 3380 to 3390). The tuning heading will include the new device name (e.g. CBL TUNED FOR 3390). The device type can be any of the following DASD:
   0671 3330 3340 3350 3380 3390 3310 3370 3375 9332 9335 9336 9345

3. CYLMAX=nnnn
   CYLMAX=nnnn is used to limit the maximum size of an allocation. The default is the device capacity of the largest type of either the device specified or the current device.

On-line and Batch files

Tuning is usually carried out on a particular file/set of files. It is possible to On-line and batch files have specific access requirements which should be reflected in the DEFINE parameters.

On-line Files

These are files which are used by several applications concurrently and whose access is normally random. They should be tuned when the file is closed to the on-line environment (e.g.CICS), as Catalog statistics are not updated until file
For on-line files a small CI size is most efficient as it keeps buffer space storage requirements to a minimum (As these files are accessed randomly, it is unlikely that 2 consecutive I/Os would involve the same CI, therefore a CI size which can contain a large number of records is unnecessary).

**Batch Files**

These files are normally accessed sequentially. In this case a large CI size is most efficient, as it keeps I/O to a minimum (The transfer of a large number of records in a single I/O is an advantage if all the records will be required).

It is possible to tune on-line and batch files within a single run. However, for greater control, they should be tuned in separate operations. The sub-parameters related to on-line and batch file tuning are described below:

1. **CBLVCONL=X’xx’**  
   Specifies which character strings (within the filename) activate on-line tuning for that file.
   
   A filename is considered to be comprised of "qualifiers" separated by periods ("."). Only strings that match complete qualifiers within the filename will activate the on-line tuning. (If this method does not comply with your installation standards, see **KEY** and **IGN** for alternative methods of file selection).

2. **CBLVCONS=xxx**  
   Specifies the user defined character. string (within the filename) that sets on-line tuning (This string is used only if bit X'01' of CBLVCONL is set on).

3. **CBLVCONT=nnnn**  
   The maximum CI size allowed for on-line tuning (default 4096).

4. **CIMAX=nnnn**  
   Used to limit the maximum CI size selected (particularly useful for on-line files).

5. **CIMIN=nnnn**  
   Used to limit the maximum CI size selected (particularly useful for batch files).

6. **CISIZE=n1,n2**  
   This is the same as specifying both **CIMIN=n1** and **CIMAX=n2**.

**Alternate Index Files**

The default values (i.e. those allocated by VSAM if no values were supplied at DEFINE time) for average and maximum record sizes are 4086 and 32600 respectively. Unless these are truly representative, allowing CBLVCAT to use these default values will result in over allocation at tune time (A WARN 014 message is produced for this condition). Inaccurate record sizes can be overridden using the **AVLRECL** and/or **MAXLRECL** parameters.

**Database Files**

The default values (i.e. those allocated by VSAM if no values were supplied at When tuning Database files, users may want to preserve the CI size by specifying the parameter **CISIZE=KEEP**. This is because any alteration in CISIZE can adversely affect LSR buffer pool definitions. See Tuning for IMS/DL1 Databases in section Technical Information.

If Catalog statistics (e.g. **Record Count**) are not maintained by VSAM, tuning could be based on inaccurate values. The CBLVCAT tuning overrides (e.g. **RECORDS**) are used to replace the innaccurate values with accurate ones.

**Reusable Files**

At tuning time (unless the **RECORDS** parameter is used as an override) the space allocation is based on the maximum capacity of the current total allocation, rather than on the Catalog record count (which may not contain a representative value if CBLVCAT is run at a non-peak time of the file's cycle).

**Average Record Length Estimation**

File fragmentation, caused by CI or CA splits occuring at record insertion time, may result in CBLVCAT being unable to judge the extent of freespace distribution. This could lead to CBLVCAT basing its tuning recommendations on an incorrect estimate of the Average Record length. A file reorganisation (BACKUP/*DELETE/*DEFINE/*RELOAD) will overcome this problem (See the IDCAMS DEFINE and Reorganisation section) Alternatively, SELCOPY can be used to read the whole file and calculate the precise Average Record Length, which can then be supplied to CBLVCAT via the **AVLRECL** parameter (contact the SELCOPY query desk if assistance is required for this exercise).

It should also be remembered that the average record length estimation may be affected by inaccurate Catalog statistics (possibly due to open files or previous abnormal program terminations). Running CBLVCAT when the file is not open for update by another
program (and after an IDCAMS VERIFY in the case of an abend) will overcome these situations.

CBLVCAT’s estimated average record length has to be +/-50% of the defined value to cause a change to be recommended. However, if other file characteristics have initiated tuning recommendations, the estimated value will be reported whatever its variance from the defined value. (A WARN 015 message is produced when the estimated value is +/-50% of the defined value).

**Growth and Freespace**

CBLVCAT normally recommends a primary allocation of sufficient size to hold all the existing records. This allocation usually has space for additional records, as it is rounded up to the next efficient boundary (e.g. to a full cylinder). However, for reusable files the capacity of the current prime allocation is used instead. The TUNE sub-parameters which relate to growth and freespace are as follows:

1. **GROWTH=nn**  
   KSDS/AIX files  
   The percentage of space to be reserved, at initial load, for insertions. The free space parameters (bytes/CI and CIs/CA) are calculated assuming that inserts occur uniformly throughout the file (Use the RECORDS parameter to influence the initial load size). If GROWTH is used, FRSPCA and FRSPCI and cannot be used.

2. **FRSPCA=nn** (KSDS/AIX only).  
   Used to specify the percentage of control intervals within a CA which are to be reserved to accommodate CI splits. Use of this parameter (and FRSPCI) results in an absolute value for the freespace in a CA (or CI with FRSPCI), rather than a more general specification based on projected file growth.

3. **FRSPCI=nn** (KSDS/AIX only)  
   Used to specify the percentage of free space within a control interval, reserved to accommodate record insertions.

4. **RECORDS=nnn**  
   KSDS/AIX files  
   Specifies the number of records required at initial load (Use the GROWTH parameter to influence the space reserved for insertions).

   ESDS/RRDS files  
   Used to change the capacity of the prime allocation. If not specified, the existing file size is used, except for Reusable files, where the capacity of the current prime allocation is used.

---

**Tuning Output**

1. Introduction  
2. SEVerity Block  
3. TUNE Block  
4. CAPacity Block  
5. JCL Override Block  
6. Example 22. Increased Initial Load  
7. Example 23. Limiting CI Size  
8. Example 24. Conditioning Distributed Freespace  
9. Example 25. Conditioning Absolute Freespace

---

**Introduction**

For each file tuned, the tuning information follows the standard report information (see Guide to LIST Output for non-tuned output). The tuning information is supplied via one, or more, of the four available “tuning blocks” (The number of blocks displayed depends upon installation defaults and run-time options).

See Example 20 in Guide to LIST Output for information regarding the production of the tuned IDCAMS DEFINE parameters to SYSPCH/SYSPUNCH.

The following example shows the tuning layout (all four blocks are displayed).
Each block's format and content is discussed in detail in the four sections which follow:

**SEVerity Block**

This is the first tuning block and contains the file Severity Messages. (See SEV Messages for a detailed description of all SEVerity messages). It may be suppressed with the NOPSEV option and may also be printed for a non-tuning run using the PRTSEV option.

The severity messages are CBLVCAT's method of indicating problem files. The level (1-3) of the message indicates how severe CBLVCAT thinks the file problems are.

| SEV 3-nn | Files should be tuned and redefined immediately |
| SEV 2-nn | Files should be changed when next loaded     |
| SEV 1-nn | Files have characteristics which require investigation |

The SEV messages from the above tuning output example are as follow:

*** SEV 3-03 *** ACTUAL FRSP CI = 0 PC
** SEV 2-04 ** BUFSP TOO SMALL FOR EFFICIENCY
** SEV 2-25 ** INEFFICIENT DATA CISIZE
* SEV 1-15 * 9 CYLS CAN BE RECOVERED WHEN TUNED

There will always be at least one SEV 2-nn and/or SEV 3-nn message for each file in need of tuning, otherwise, by definition, the file is already in tune. If tuning has been influenced by sub-parameters, additional appropriate SEV 2-nn messages are displayed, showing why the file has been tuned. A warning message can also follow this severity message block which relates either to the tuning recommendations (e.g. A change to the allocation of +/-50% or more - WARN 016 as in the example), or to the tuning request (e.g. WARN 011 NO FILES TUNED).
TUNE Block

This is the first tuning block and contains the file Severity Messages. This is the second tuning block and contains the recommended tuned IDCAMS DEFINE parameters. It cannot be suppressed for a tuning run, however, if CBLVCAT considers the file to be in tune (i.e. No SEV 2-nn or SEV 3-nn messages), the detail will be suppressed and the message "== FILE ALREADY IN TUNE ==" will be displayed (If all files in the tuning run were considered to be in tune a WARN 011 message would also be included after the SEV block). There may be SEV 1-nn messages but these would not be reflected in the DEFINE parameters, which will be unchanged from the current definition.

If a file is selected for tuning (i.e. SEV 2-nn or SEV 3-nn messages exist) all SEV message recommendations (including SEV 1-nn) will be incorporated into the tuned DEFINE parameters.

If tuning overrides have been supplied this automatically invokes a SEV 2-nn message causing the DEFINE parameters to incorporate all SEV changes, even if the file would otherwise have been considered in tune (i.e. if the only other messages were SEV 1-nn these would be incorporated into the revised DEFINE parameters).

The tuning block has two possible headings:

1. CBL TUNED
   If the DEV parameter was omitted from the tuning run (i.e. tuning was for the current device).

2. CBL TUNED FOR DEV=nnnn
   If DEV=nnnn was specified (Indicating that the tuning output is based on the device specified as the argument.

Because of the inherent danger of defining parameters at the CLUSTER level (e.g. a single CI size applying to both data and index components would almost certainly overallocate the index), CBLVCAT's tuning output is divided into two sections with headings as follows:

1. DATA
   Subsequent tuning recommendations apply only to the data component of the cluster (The final recommended change will be followed by a closing bracket '), indicating the end of data component recommendations).

2. INDEX
   Subsequent tuning recommendations will apply only to the index component of the cluster (The final recommended change will be followed by a closing bracket '), indicating the end of index component recommendations). Most INDEX values should be left for VSAM to select default values.

Tuning recommendations from the above tuning output example are as follow:

```
CBL TUNED
--------
DATA (                           - * NEW PHYREC SIZE=14336, CURRENT=4096
  CISZ (14336) - * OPTIMISED FOR DEVICE GEOMETRY
  CYLINDERS (3,1) - * DEFINED AVLRECL=700
  RECORDSIZE (776,4089) - * GIVES FREE REC=2/18, FREE CI=2/42 (IMBED)
  FREESPACE (9,3) - * 34K MINIMUM FOR DIRECT PROCESSING
  BUFFERSPACE (34816) )
```

All possible recommendations which apply to the DATA component follow:

**CISZ(nnnnn)**
Can be included for both the INDEX and DATA components (INDEX recommendations are discussed below).

Its presence indicates that a change to the data component CI size (control interval size) value is being recommended (See CI size in Additional VSAM Information). The value is subject to any restrictions made via the CIMIN/ CIMAX/ CISIZE or the CBLVCONL/ CBLVCONS/ CBLVCONT (on-line selection) tuning sub-parameters. It is displayed in one of the following formats:

1. **CISZ(nnnnn)**
   Recommends the CI size which will best utilise the average record length and device characteristics

2. **CISZ(nnnnn) - * ONLINE VALUE**
   The file is an on-line file, selected in accordance with the CBLVCONL and CBLVCONS options (or by the appropriate CBLNAME values). The maximum CI size available is defined in the CBLVCONT option (or CBLNAME value) and defaults to 4096. (See description in On-line and Batch files earlier in this section).

3. **CISZ(nnnnn) - * NEW PHYREC SIZE=nnnn, CURRENT=nnnn**
   For CKD devices, this indicates that the recommended CI size will result in a change to the physical record size used by VSAM. It normally occurs when tuning for a different operating system (TUNE sys), but can occur for files that were defined by previous versions of VSAM (FBA devices always have 512 byte physical records).

4. **CISZ(nnnnn) - * NEW PHYREC SIZE=nnnn, CURRENT=nnnn (DEV CHANGE)**
   This advises that the CI size change may be caused by the device change (The current CI size may be correct for the current device).

**BLOCKS (nnn nnn) or CYLINDERS (nn n) or TRACKS (nnn nn)**
Specifies the disk space to be allocated to the file. For FBA devices the values will be displayed in blocks, whereas for CKD devices the values will be displayed in cylinders (except for very small files which will be displayed in tracks).
The first value displayed refers to the size of the **Primary Allocation** which is space reserved at load time.

The second value refers to the size of the **Secondary Allocation**. Space for a Secondary Allocation is not reserved at load time but is acquired each time the current allocation is full and a further record addition takes place. (Of course, if the file is defined with an insufficient amount of primary allocation, secondary allocation could be acquired at load time).

A **WARN 016** message is produced for allocation recommendations of +/-50% of the current value.

A **WARN 017** message is produced when the estimated number of records (e.g. for a reusable file), is +/-50% of the current number of records. This also affects the primary and secondary allocations.

The allocation recommendation may be accompanied by one of the following comments:

1. **OPTIMISED FOR DEVICE GEOMETRY**
   The recommended allocation has been increased because of the minimum CA/maximum CA (track/cylinder) relationship of the device. This is because processing is most efficient if allocations are either in full cylinders, or a factor of a cylinder.
   e.g. If the allocation is currently 4 tracks, a change to 5 will be recommended for a device with 15 tracks/cylinder (CBLVCM always 'rounds upwards').

2. **LIMITED BY CYLMAX PARM**
   The **CYLMAX=nnn** parameter has been supplied as a tuning override and the file is slightly too big to load into this restricted primary allocation. The complete file will load with one secondary allocation.

3. **LIMITED BY CYLMAX PARM (n PRIMARY VOLS REQD)**
   The **CYLMAX=nnn** parameter has been supplied as a tuning override and loading the file would require more than one secondary allocation. The file therefore needs multiple primary allocations, which requires a **VOL** entry to be included in the **DEFINE** parameters (A **WARN 018** message is also produced for this condition).

4. **LIMITED BY DEVICE CAPACITY**
   The primary allocation is limited by the device size and the file is slightly too big to load into this restricted primary allocation. The complete file will load with one secondary allocation.

5. **LIMITED BY DEVICE CAPACITY (n PRIMARY VOLS REQD)**
   The primary allocation is limited by the device size and loading the file would require more than one secondary allocation. The file therefore needs multiple primary allocations, which requires a **VOL** entry to be included in the **DEFINE** parameters.

**Note**
VSAM also allows Allocation to be specified in **RECORDS. This is not recommended** as different devices require different CA size calculations.

**RECORDSIZE(nnn nnn)**
Indicates the revised Average and Maximum record lengths. The first value displayed will relate to the average record length. If the **AVLRECL** parameter has been supplied, this will be shown as the recommended change. Otherwise, the value indicates that CBLVCM’s estimated average record length differs from that defined (For KSDS files, with distributed free space, this calculation can only be approximate - See the Average Record Length Estimation in Tuning Considerations).

A change of +/-50% will activate tuning recommendations, however, if other characteristics have already initiated tuning output, the tuned value will be reported whatever the percentage change. (A **WARN 015** message is also produced for average record length recommendations of +/-50% of the current value).

The second value displayed will relate to the maximum record length. If the **MAXLRECL** parameter has been supplied, this will be shown. Otherwise, the current maximum record length will be shown. In either case, it will have been increased to the new average record length, if it would otherwise have had a lower value.

The **RECORDSIZE** recommendation will always have the following comment:

1. **DEFINED AVLRECL=nnn**
   The defined average record length is displayed in order to show the size of the recommended change.

**Note**
The display of record lengths in the standard report is governed by the **AVRL / LMAX** option, which defaults to **LMAX**. (If set to **LMAX** there is no other indication on the report of the defined average record length value).

**FREESPACE(nn nn)** (KSDS and AIX files only)
Indicates the recommended amount of free space to be reserved for future record insertions. The recommendation will occur in one of the following forms (Note - when the **IMBED** literal is shown, the CA capacity has been reduced by one minimum CA to reflect this attribute):

1. **FREESPACE(n n) - * GIVES FREE REC=n/n, FREE CI=n/n (IMBED)**
   The two numeric values (separated by a blank), indicate the recommended percentage of freespace within Control Intervals and Control Areas respectively.
   - **FREE REC=n/n**
     Indicates the number of record slots left free in a CI, compared with its total capacity (The number of records per CI during initial load may be calculated by subtracting these two values).
FREE CI=n/n
Indicates the number of CIs left free in a CA, compared with its total capacity (The number of CIs used per CA during initial load may be calculated by subtracting the two values).

The CA capacity (during file loading) can be calculated by multiplying used records by used CIs (Other file capacity information is shown in the File Capacity block described later).

e.g. Using the figures from the previous example.

* GIVES FREE REC=2/18 FREE CI=2/42 (IMBED)

Represents (18 - 2) * (42 - 2) = 640 records per CA.

2. FREESPACE(0 n) - * GIVES FREE REC=0/n, FREE CI=n/n (IMBED)
Always occurs when FRSPCI=0 has been specified. If it has not been specified, it shows that your chosen CISZ and FREESPACE values have combined to produce an 'effective' FREESPACE of zero, i.e. The actual amount of space reserved may not be large enough to accommodate a single insertion. It is therefore more efficient to reserve no freespace, rather than unusable freespace. If this is unacceptable your CISIZE override (or FRSPCI=nn) should be reconsidered (GIVES FREE is described above).

3. FREESPACE(n 0) - * GIVES FREE REC=n/n, FREE CI=0/n (IMBED)
Always occurs when FRSPCA=0 has been specified.

In the case of small files it can also have another meaning. It shows that CBLVCAT is recommending no freespace for files of 2 CIs or less or that the percentage of freespace/CA that has been specified has resulted in a freespace of less than 1 CI, therefore CBLVCAT is recommending no freespace is reserved (GIVES FREE is described above).

4. FREESPACE(0 0) - * GIVES FREE REC=0/n, FREE CI=0/n (IMBED)
Occurs if free space exists and the GROWTH=0 parameter (or FRSPCI=0 and FRSPCA=0) has been supplied.

Note
Removing free space for a KSDS should not be an automatic change. Confirmation should first be obtained that the file will not have subsequent insertions, as they would then cause immediate CI and CA splits.

BUFFERSPACE(nnnnn)
Indicates the minimum efficient Bufferspace value for random processing

Storage allocation for Bufferspace, as well as being defined in the catalog, may be increased (but not decreased) for the duration of a job, or jobstep, using a JCL override (see JCL Overrides Block). As the value may not be decreased at run time, it is important that the defined value is the minimum required for efficient processing.

The BUFFERSPACE recommendation will always have the following comment:

1. * nnK MINIMUM FOR DIRECT PROCESSING
The value is rounded up to a 2K boundary (where K=1024). The comment is a reminder that the value relates to random processing (sequential access generally requires a larger value - see On-line and Batch Files earlier in this section)

NONSPANNED
The file has been defined with an unnecessary SPANNED attribute. A small increase in CI size has been sufficient to allow all records to fit within single control intervals.

Spanned records are inefficient as they must start at the beginning of a CI, and the CI containing the last segment cannot hold other records. CBLVCAT recommends un-spanning a file if its maximum LRECL+7 does not exceed the lower of CMAX and 5 times CI size.

The NONSPANNED recommendation will always have the following comment:

1. * OR REMOVE SPANNED
This indicates that NONSPANNED is the DEFINE default and removing the existing SPANNED parameter is sufficient.

Note
The CISIZE=KEEP parameter can be used as an override to stop CBLVCAT recommending the removal of the SPANNED attribute.

NOWRITECHECK
Defining a file with WRITECHECK adds an additional read I/O operation after each write. With the reliability of modern disk technology it is no longer necessary, therefore it is recommended that WRITECHECK is not specified.

The NOWRITECHECK recommendation will always have the following comment:

1. * OR REMOVE WCHK
This indicates that NOWRITECHECK is the DEFINE default and removing the existing WCHK parameter is sufficient.

SPEED
Shows that the file was defined with, or allowed to default to, the DEFINE attribute RECOVERY (This is a costly option as it can increase the time taken to load a file by up to 40% and is only useful if you have a load re-start procedure).
RECOVERY causes VSAM to preformat each Control Area before it loads records into it. Therefore, if the load fails, the last record written is automatically followed by an end-of-file record. The time increase that this preformatting causes for each load, far outweighs the time saving in the unlikely event of a load failing.

The SPEED recommendation will always have the following comment:

1. * DON’T ALLOW DEFAULT
   This indicates that the SPEED attribute must be specifically coded in the IDCAMS DEFINE, as RECOVERY is the default.

All possible recommendations which apply to the INDEX component follow:

CISZ(nnnnn)
This recommendation can be included for both the INDEX and DATA components (DATA recommendations are discussed above).

When included for the index component, it indicates the minimum value required assuming normal key compression. However, if your file has large keys and the front and backs of keys tend to change very frequently, the keys will not compress well. In this case you should increase the recommended index CISIZ value by up to 2048 bytes. (See key compression in section Additional VSAM Information).

The following comment may be displayed:

1. * VALUE IS NOW CORRECT WITH THE NEW DATA CISIZE
   This indicates that the current index CISIZ is correct if the recommended change to the data CISIZ is made.

NOIMBED
This change is recommended to the index component of small files.

When a KSDS file is defined with the IMBED option, the index sequence set is moved to the first track of each CA and repeated as many times as it will fit on the track. For large heavily used files (especially those used on-line) IMBED can help in two ways. It can reduce the amount of index I/O activity and it can reduce disk rotational delay which results in faster transfer of the required CI. It is of little or no value to small files.

The NOIMBED recommendation will always have the following comment:

1. * OR REMOVE IMBED
   This indicates that NOIMBED is the DEFINE default and removing the existing IMBED parameter is sufficient.

CAPacity Block
This is the third tuning block and shows capacities (in records) of the primary and secondary allocations for the tuned and current values (A WARN 017 message is produced when the estimated number of records is +/-50% of the current NRECS value). It can be suppressed using the NOPCAP option.

The original file capacities can have 2 different headings, these are:

1. --- ORIG (ESTD) ---
   This is the heading if the file has fixed length records and CBLVCAT knows the exact length.

2. ----- CURRENT ----- 
   This is the heading if CBLVCAT is using its estimate of the average record length.

The block format depends upon whether the files are KSDS/AIX or ESDS/RRDS.

KSDS files

<table>
<thead>
<tr>
<th>USING AULRECF=150</th>
<th>--- FILE CAPACITIES (NRECS) ---</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>---</td>
</tr>
<tr>
<td>*</td>
<td>PCNT PRIME</td>
</tr>
<tr>
<td>* INIT LOAD</td>
<td>79 504K</td>
</tr>
<tr>
<td>* FREE</td>
<td>21 134K</td>
</tr>
<tr>
<td>*</td>
<td>100 639K</td>
</tr>
</tbody>
</table>

The percentages illustrate the number of prime records achieved out of the maximum possible primary allocation (Assuming the optimum CI size for the device). This example is based on a 3390 and so uses a data CI size of 18K (for a 3380 use 22K). The allocation capacities are calculated by dividing the track capacity by the average record length.

It is important to realise that the actual number of records in the original and tuned sections cannot be compared directly. This is because CBLVCAT may have also recommended changes to the allocation size, which would result in a
comparison between values based on different allocation sizes. The important figures for comparison are the percentages, which show the relative success in utilisation of the primary disk allocation. In the above example:

◊ 639K records represent 100% of the total primary allocation capacity.
◊ 504K records represent 79% of the total primary allocation capacity.
◊ 134K records represent 21% of the total primary allocation capacity.

IMBED uses one track (or minimum CA) per data CA for index, which reduces the percentages accordingly.

**ESDS/RRDS files**

```
== FILE ALREADY IN TUNE ==
* USING AVLRECL=100    --- FILE CAPACITIES (NRECS) ---
*             ------ TUNED ------           --- ORIG (ESTD) ---
*             PCNT   PRIME    SEC           PCNT   PRIME    SEC
*             ----   -----    ---           ----   -----    ---
*              84     4680    360                == SAME ==
```

ESDS and RRDS files do not have free space reserved during initial load.

The information displayed in the block has the same meaning as for KSDS files except that the concept of free space does not exist. Therefore, there is only one line displayed for the initial load.

The example above shows the block when a file is already in tune (The "FILE ALREADY IN TUNE" message can be shown even when there is a new estimated average record length which is not +/- 50% of the defined value).

**JCL Override Block**

This is the fourth and last tuning block and contains the suggested JCL overrides for enhanced **sequential** processing. It can be suppressed using the **NOPJCL** option.

**BUFND** values are supplied for MVS systems, and VSE/SP systems from 1.4.1, **BUFSP** for VSE systems prior to 1.4.1. (**BUFSP** is rounded to the next highest 2K boundary).

Examples of both types follow:

**OPTIONAL JCL OVERRIDE FOR SEQ I/O**
```
BUFND=5      * (70K)  FAST
BUFND=8      * (112K) URGENT
BUFND=25     * (350K) **TURBO**  (BUT WILL IMPACT OTHER WORK)
```

Example: Tune JCL - BUFND (CBL Ref: vmxjclo)

**OPTIONAL JCL OVERRIDE FOR SEQ I/O**
```
BUFSP=73728  * (72K)  FAST
BUFSP=116736 * (114K) URGENT
BUFSP=360448 * (352K) **TURBO**  (BUT WILL IMPACT OTHER WORK)
```

Example: Tune JCL - BUFSP (CBL Ref: vmxjcid)

The 3 recommended JCL overrides are described as follows:

1. **FAST**
   
The value required for normal fast sequential processing (The additional buffers will not normally impact other users).

2. **URGENT**
   
   Use this value for high priority jobs (The additional buffers will have some effect on other users). If all users submit URGENT jobs, the advantage over FAST is lost and additional memory has been paged in for no effect.

3. **"TURBO"**
   
   To be used in exceptional cases only, particularly if the (BUT WILL IMPACT OTHER WORK) comment is displayed (The additional buffers will have a large effect on other users, but will let your top-priority job finish extremely quickly).

It is also possible (for small files) for additional information to be printed alongside the above messages. These are warning indications about the bufferspace implications relating to small file sizes. The possible messages are:

1. **(THIS IS ALREADY THE DEFAULT)**
   
The defined BUFSP (for random processing) is already sufficient for this level of sequential processing. This will occur for files with a small maximum record length and data CI size, and so require a relatively large index CI size.

2. **"DO NOT USE" (FILE TOO SMALL)**
   
The number of control intervals actually containing data is lower than the number of buffers that would be allocated for this level of processing. Storage would therefore be allocated, but never used.
Example 22. Increased Initial Load

This example illustrates tuning with an increased. initial load capacity requested by the RECORDS sub-parameter.

e.g.

```cbl
LC DD=CB9009   KEY=/SUMM
TUNE  RECORDS=500000
NOASSOC NOPJCL NOVOL        * OPTION parameters
```

The message completely replaces the particular bufferspace recommendation to which it applies and means that the primary allocation can be read, in full, into a smaller buffer area. Application of this particular recommendation would have resulted in over-allocation i.e. unusable storage would have been reserved.

Notes

1. The file selected for tuning has a Data Set Name containing the string 'SUMM' from a Catalog referenced by CB9009.

2. The JCL overrides block has been suppressed using option NOPJCL.

3. Volume information has been suppressed using option NOVOL.

4. Association information has been suppressed using option NOASSOC.

5. The record count from the Catalog is superseded by the value given with the RECORDS parameter (in this case 500000).

6. The data CI size increases from 4096 to a more efficient 18432.

7. The Allocation increases from CYL(50 50) to CYL(120 15) in order to accommodate the revised primary load requirement (i.e. 500000 records).

8. FREESPACE is adjusted for Control Interval and Control Area.

9. The Index CI size is reduced from 4096 to 1536, the minimum required assuming normal key compression. (See KEY COMPRESSION in Additional VSAM Information).
10. See the earlier SEVerity block description for an explanation of the SEVerity and WARN 017 messages.

Example 23. Limiting CI Size

This example illustrates the effect of supplying a CIMAX parameter, typically to limit CI size for on-line use.

e.g.

```
LC DD=CBL111 KEY=CBL.PROD.INVOICE.LINE
TUNE CIMAX=4096
NOVOL NOASSOC
```

Example: Limiting CI Size (CBL Ref: vmxtex3)

Notes

1. The file selected for tuning has a Data Set Name commencing with the string 'CBL.PROD.INVOICE.LINE' from a Catalog referenced by CBL111.
2. Volume information has has been suppressed using option NOVOL.
3. Association information has been suppressed using option NOASSOC.
4. A CI size upper limit of 4096 is imposed via the CIMAX parameter, as the file is used predominately in an online environment.
5. The Allocation is optimised for the device geometry of a 3380.
6. The Index CI size is increased from 1024 to a more efficient 2048.
7. Bufferspace is reduced in line with the changes in Data and Index CI size.
8. See the earlier SEVerity block description for an explanation of the SEVerity messages.
### Example 24. Conditioning Distributed FREESPACE

This illustrates the **GROWTH** parameter. The numeric value specified indicates the percentage increase expected in the file size (**RECORDS** is mentioned in example 22).

```
TUNE GROWTH=20 RECORDS=9999
```

e.g.

```
LC DD=CBLV05 KEY=ACCTS.TEST.SORTED TYPE=K
```

---

#### Notes

1. The file selected for tuning is a KSDS file whose Data Set Name commences with the string 'ACCTS.TEST.SORTED'. It is selected from a Catalog referenced by CBLV05.

2. A **distributed** free space allowance equal to 20% of the load size, has been requested via the **GROWTH** parameter.

3. The **RECORDS** parameter requests a primary allocation large enough to accommodate 9999 records, compared with the current record count of 7400.

4. The Data CI size increases from 4096 to a more efficient 6144.

5. The Allocation increases from **TRK(40 8)** to **CYL(6 1)**, to accomodate the supplied **GROWTH** and **RECORDS** requirements.

6. The Index CI size is increased from 1024 to a more efficient 1536.

7. See the earlier SEVerity block description for an explanation of the SEVerity and **WARN 016** messages.
Example 25. Conditioning Absolute FREESPACE

This example illustrates the setting of the absolute free space parameters FRSPCI and FRSPCA. Under normal circumstances the GROWTH parameter would be used to specify file growth. This is because FREESPACE specification using these two parameters, requires the user to reference file statistics for accurate results. They do, however, provide more control by allowing separate specification of the FREESPACE within the CIs and CAs, as opposed to general file growth.

e.g.

```
LC DD=CBLI11 KEY=/SUMM
TUNE FRSPCI=10 FRSPCA=5
```

Example: Absolute Freespace (CBL Ref: vmxtex6)

Notes

1. The file selected for tuning has a Data Set Name containing the string ‘SUMM’ from a Catalog referenced by CBLI11.
2. Absolute free space is requested as 10% of each CI and 5% of each CA, via the FRSPCI and FRSPCA parameters.
3. The Data CI size increases from 4096 to a more efficient 18432.
4. The Allocation is optimised for the disk device and reduced FREESPACE values, recovering 22 cylinders.
5. The Index CI size is reduced from 18432 to 1536, recovering 4096 cylinders.
6. The Data CI size increases from 512 to 18432.
7. The Allocation is optimised for the disk device and reduced FREESPACE values, recovering 22 cylinders.

Example: Absolute Freespace (CBL Ref: vmxtex6)
VSAM Monitoring

1. Regular Monitoring
2. Fine Tuning

Monitoring with CBLVCAT is incredibly simple as the SEV parameter can be used to restrict reports to files which are 'out of tune' (e.g. SEV=3 reports on major problems only). To ask at the same time for tuning recommendations will introduce a negligible CPU overhead, so the TUNE parameter is also coded.

It is recommended you establish a regular monitoring interval, preferably just before the majority of your files are backed up prior to re-load, as this is generally when the files will be in their worst state of tune.

e.g.

```
LC DD=USERCT SEV=3 TUNE * Tune for problem files.
```

Regular Monitoring

In order to keep the performance gains achieved by CBLVCAT's tuning process, a regular job with subset SEV=3 or SEV=2 should be set up, using IGN parameters to bypass files that are not required (The latter category will probably consist of files that have not yet been tuned, together with those out-of-tune files whose size, or frequency of use, does not warrant re-defining and re-loading). This type of selection will produce a report limited to files which are starting to drift out of the optimal state of tune.

e.g.

```
LC DD=UCAT01
KEY=/GENERAL
SEV=2
* Select catalog.
* Choose files with GENERAL in name.
* Report problem files.

LC DD=UCAT01
IGN=ONLINE.CTL
KEY=ONLINE
KEY=ACCOUNT
LOCISZ=1024
HICISZ=4096
SEV=2
* Select same catalog.
* Ignore files starting with ONLINE.CTL
* Choose files starting with ONLINE
* Choose files starting with ACCOUNT
* Set low cisize limit.
* Set high cisize limit.
* Report problem files.
```

Fine Tuning

Once the bulk of your files are in reasonable shape, you may wish to turn your efforts to precise tuning in order to squeeze the most out of those files you consider critical. Once the files have been identified, run your tuning jobstreams at SEV=1, looking for the slightest imperfection.

You may be tempted to include fine tuning as part of the regular monitoring exercise. This is not recommended as tuning would always be recommended, even for minute file changes. The fine tuning exercise works best when directed at a few specific files at a time.

The following example may be used as a basis for developing a fine tuning job.

```
LC DD=UCAT01
KEY=FAST.BATCH
SEV=1
TUNE
CIMIN=4096
RECORDS=20000
GROWTH=15
* Select specific file for batch.
* Report minor conditions.
* Give fine tuning recommendations.
* Restrict MINCI.
* Set normal record count.
* Set expected growth from normal.

LC DD=UCAT01
KEY=ONLINE.MASTER
TYPE=KA
SEV=1
TUNE
CISIZE=KEEP
FRSPCI=10
FRSPCA=30
DEV=3380
* NEXT FILE - NEXT REPORT.
* Select specific file for on-line.
* Select KSDS and AIX.
* Report minor conditions.
* Give fine tuning recommendations.
* Don't change CISIZE.
* Specify few evenly distributed inserts.
* Lots of pocket type inserts.
* Give changed recommendations.
```

Reference Guide to VSAM Tuning and A-Z Reference, for explanations of the parameters which will help you achieve specific tuning objectives.

Another item which is vital to achieving optimum performance from your fine-tuned files, is the proper use of JCL BUFS* (BUFND and BUFNI if possible), JCL B*SP, ensures proper use of buffers for the mode of processing required by application programs. A combination of fine tuning and proper JCL BUFS* overrides will yield optimum performance (See BUFFERSPACE in Additional VSAM Information).
VSAM Modelling

1. Introduction
2. Example 26. Modelling a File
3. Example 27. Modelling for DASD Change

Introduction

CBLVCAT can be used to produce an output file containing the tuned IDCAMS define parameters based on a current file. This means that:

1. You don’t have to go through the frustrating exercise of manually analysing file requirements for future projects. Just base your definitions on a file which resembles the required profile and add tuning parameters as required.
2. You can plan DASD requirements as soon as the decision to convert is made, not after the DASD have arrived.
3. You can plan Operating System conversions, with or without different DASD, long before you start experimenting with the new system, or before the related manuals arrive.

As modelling is really only a term used to show a particular use of file tuning, all relevant information is covered in the sections Tuning Considerations and Tuning Output. The following 2 examples illustrate the concept.

Example 26. Modelling a File

This example illustrates modelling an ESDS file.

e.g.

```
LC DD=CBLI11 KEY=CDBPV.AP.DATE.MASTER DEFINE
TUNE AVLRECL=500 MAXLRECL=500 RECORDS=15000
```

Example: Modelling an ESDS file (CBL Ref: vmxtex2)
The Data Cl size increases from 8192 to 18432 in line with the supplied average record length.

Allocation is optimised for the characteristics of the disk device. This gives CYL(10 2) instead of TRK(7 1).

The file CDBPV.AP.DATE.MASTER from the Catalog CBL11 is used as a model, with override values for RECORDSIZE (via AVLRECL=500 and MAXLRECL=500) and allocation (via RECORDS=15000).

**DEFINE** is specified in conjunction with **TUNE** to produce an output file, to SYSPCH/SYSPUNCH, containing the tuned IDCAMS **DEFINE** parameters (see IDCAMS DEFINE and Reorganisation in Guide to LIST Output for full details).

Example 27. Modelling for DASD Change

Illustrates tuning a file for a different disk device. You are able to prepare for migration before you have the new DASD and/or operating system installed (to model for a different operating system simply specify **TUNE sys**).

e.g.

```
LC DD=CBLV10 KEY=PRODUCT.FS.PARAMETER DEFINE
TUNE DEV=3390
```

```
TUNE  DEV=3390
LC DD=CBLV10   KEY=PRODUCT.FS.PARAMETER   DEFINE

CBL TUNED FOR DEV 3390
```

**Notes**

The file **PRODUCT.FS.PARAMETER** from the Catalog **CBL10** is used as a model and tuned for a new device (3390) using the **DEV** parameter.

**DEFINE** is specified in conjunction with **TUNE** to produce an output file, to SYSPCH/SYSPUNCH, containing the tuned IDCAMS **DEFINE** parameters (see IDCAMS DEFINE and Reorganisation in Guide to LIST Output for full details).

The Data Cl size increases from **4096** to a more efficient **18432**.
The Index CI size increases from 512 to 1024 to eliminate inefficiency.

Allocation is optimised for the characteristics of the new disk device giving CYL(1 1) instead of TRK(3 3).

See the SEVerity block description in Tuning Output for an explanation of the SEVerity messages and warning message WARN 016.
# Summary of Syntax

In the following tables the syntax of separating keywords with a vertical bar ("|") is used to indicate that the keywords are mutually exclusive.

## REPORT Summary

<table>
<thead>
<tr>
<th>Command</th>
<th>**</th>
<th>Common Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>BLKSIZE DSN TYPE VOLUME</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CISIZE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VCAT only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALLOCP ENTRY NRECS SMS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALLOCS EXCPS NSRC SMSD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALLOCT FREEBYTES PCNT SMSS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALLOCU FRSP PHYREC SMSS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALLOC3 GGEN RECDEL SPLITCA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALLOC4 GMAX RECINP SPLITCI</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ASSOC GVER RECINS SPLIT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AVRL HIALLRBA RECSTATS TIME</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BUFSP HIUSERBA RECUPD VOL1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BUFSP/IXL IMB RKP VOL2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CATALOG IXL S/C VOL3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CI/CA KL SEVL VOL4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>COMPONENT KL/BLK/IMB SHR VOL5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFINED LMAX</td>
</tr>
<tr>
<td></td>
<td></td>
<td>VTOC only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ACCESSED CYL/HD LRRCL USED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALLOC EXPRES REC FM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CREATED INFO START</td>
</tr>
</tbody>
</table>

### Notes

1. See **REPORT** in the A-Z Reference.

2. **SORT** and **SORTD** are special report parameters as they are *positional keywords*. They effect a **SORT** (descending in the case of **SORTD**) based on the parameters which follow them in the list. They can therefore appear anywhere within a **REPORT** command parameter list.

3. *The **STOPAFT** parameter is used to limit print output and can only be used in conjunction with the **SORT** and **SORTD** parameters.*
### OPTION Summary

<table>
<thead>
<tr>
<th>Command</th>
<th>Common Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBLCLNL=nn</td>
<td>CBLVCPCF=nn</td>
</tr>
<tr>
<td>CBLVCALT=nn</td>
<td>CBLVCFCT=nn</td>
</tr>
<tr>
<td>CBLVCLNL=nn</td>
<td>CBLVCFCV=nn</td>
</tr>
<tr>
<td>CBLVCEXT=nn</td>
<td>CBLVCRCM=X'xx'</td>
</tr>
<tr>
<td>CBLVCFN=xxx</td>
<td>CBLVCSGA=nn</td>
</tr>
<tr>
<td>CBLVCONL=X'xx'</td>
<td>CBLVCSCI=nn</td>
</tr>
<tr>
<td>CBLVCNS=xxx</td>
<td>CBLVCSAP=n</td>
</tr>
<tr>
<td>CBLVCNT=nnnn</td>
<td>CBLVCSW8=X'xx'</td>
</tr>
<tr>
<td>CBLVCSW9=X'xx'</td>
<td></td>
</tr>
</tbody>
</table>

#### LISTVCAT only

<table>
<thead>
<tr>
<th>option</th>
<th>meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALIAS</td>
<td>NOALIAS</td>
</tr>
<tr>
<td>ASSOC</td>
<td>NOASSOC</td>
</tr>
<tr>
<td>DEFINE</td>
<td>NODEFINE</td>
</tr>
<tr>
<td>EXCPS</td>
<td>CI/CA</td>
</tr>
<tr>
<td>GDDSRM</td>
<td>NOGDDSRM</td>
</tr>
<tr>
<td>GDGRPT</td>
<td>NOGDGRPT</td>
</tr>
<tr>
<td>EXCP</td>
<td>CI/CA</td>
</tr>
<tr>
<td>DEFINE</td>
<td>NODEFINE</td>
</tr>
<tr>
<td>ASSOC</td>
<td>NOASSOC</td>
</tr>
<tr>
<td>ALIAS</td>
<td>NOALIAS</td>
</tr>
</tbody>
</table>

#### LISTVTOC only

<table>
<thead>
<tr>
<th>option</th>
<th>meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOFREE</td>
<td>FREE</td>
</tr>
<tr>
<td>FREE</td>
<td>FREETAB</td>
</tr>
<tr>
<td>EXPD</td>
<td>NOEXPD</td>
</tr>
<tr>
<td>PERM</td>
<td>NOPERM</td>
</tr>
</tbody>
</table>

#### OPTIONS

- GDGRPT, NOGDGRPT, PCNT, UNUSED, VOLINFO
- OVLAY, NOOVLAY, TIMESTAMP, DEFINED
- CI/CA, MOUNT, NOMOUNT, SHR, S/C
- ISC, NOISC, PRTJCL, NOPJCL
- AVRL, PRTSEV, NOPSEV
- NMAX, AVRL, PRTSEV, NOPSEV
- LOCISZ=cisize, HICISZ=cisize
- SEV=n, SPLIT=n
- SPANNED, NRECS=nnn
- TOTLOC=nnn, SECALLOC=nnn
- TYPE=xxx, VOL=volser
- UNALLOC=nnn, CLASS=n
- COMPRESSED, EXT-ADDR
- EXTENDED, STRIPED
- XVSAM

### LISTVCAT Summary

<table>
<thead>
<tr>
<th>option</th>
<th>meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Any of the applicable options from the OPTION table.</td>
<td></td>
</tr>
<tr>
<td>LOBXL, LOCYL, LCHAIN=nnn</td>
<td></td>
</tr>
<tr>
<td>HIBXL, HICYL, HCHAIN=nnn</td>
<td></td>
</tr>
<tr>
<td>KEY=string, KEY=&quot;string&quot;</td>
<td></td>
</tr>
<tr>
<td>IGN=string, IGN=&quot;string&quot;</td>
<td></td>
</tr>
<tr>
<td>LODATE=string, HIDATE=string</td>
<td></td>
</tr>
<tr>
<td>LOCISZ=cisize, HICISZ=cisize</td>
<td></td>
</tr>
<tr>
<td>SEV=n, SPLIT=n</td>
<td></td>
</tr>
<tr>
<td>SPANNED, NRECS=nnn</td>
<td></td>
</tr>
<tr>
<td>TOTLOC=nnn, SECALLOC=nnn</td>
<td></td>
</tr>
<tr>
<td>*TYPE=xxx, VOL=volser</td>
<td></td>
</tr>
<tr>
<td>UNALLOC=nnn, CLASS=n</td>
<td></td>
</tr>
<tr>
<td>COMPRESSED, EXT-ADDR</td>
<td></td>
</tr>
<tr>
<td>EXTENDED, STRIPED</td>
<td></td>
</tr>
<tr>
<td>XVSAM</td>
<td></td>
</tr>
</tbody>
</table>

### Notes

1. See OPTION in the A-Z Reference.
1. See **LISTVCAT** in the A-Z Reference.

2. **OPTION**, **SUBSET** and **TUNE** can all be specified on the same **LISTVCAT** operation.

---

**LISTVTOC Summary**

<table>
<thead>
<tr>
<th></th>
<th>VSE only</th>
<th>* SORT</th>
<th>DATE</th>
<th>DSN</th>
<th>EXP</th>
<th>SIZE</th>
<th>OPTION</th>
<th>Any applicable OPTIONS from the OPTIONS table.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DEV=xxx</td>
<td>(VOL=vser)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>or</td>
<td>(OWN=ownr)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LISTVTOC</td>
<td>VOL=vser</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LISTV</td>
<td>MVS only</td>
<td>(FAIL=xxx EJECT MERGE)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DDNAME=fname</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>or</td>
<td>(OWN=ownr)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>_VOLUME=volser</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Notes</td>
<td>1. See <strong>LISTVTOC</strong> in the A-Z Reference.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. <strong>SORT</strong>, <strong>OPTION</strong> and <strong>SUBSET</strong> can all be specified on the same <strong>LISTVTOC</strong> operation.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. * Only one of these sort fields can be specified on a <strong>LISTVTOC</strong> operation.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**LISTLABL Summary**

<table>
<thead>
<tr>
<th></th>
<th>VSE only</th>
<th>LISTLABL</th>
<th>* No parameters, OPTIONs or SUBSETs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LISTL</td>
<td>LL</td>
<td></td>
</tr>
</tbody>
</table>

Notes

1. See **LISTLABL** in the A-Z Reference.

---

**VTOC Modification Summary**

<table>
<thead>
<tr>
<th></th>
<th>VSE only</th>
<th>DEL D</th>
<th>DEV=cuu</th>
<th>ALLFILES=BOOL</th>
<th>EJECT</th>
<th>LIST=YES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MOD M</td>
<td>VOL=volser</td>
<td>DSN=xxx.xx</td>
<td>NEWDSN=xxx.xx</td>
<td>NEWVOL=volser</td>
<td>NEWOWN=xxx.xx</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>EXP=TEMP</td>
<td>EXP=PERM</td>
<td></td>
<td>CANCEL</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OWN=xxx</td>
<td>FAIL=IGNORE</td>
<td>EOJ</td>
<td></td>
</tr>
</tbody>
</table>
Notes

1. See **DEL** and **MOD** in the A-Z Reference.
2. If **ALLFILES=YES** is used, the **VOL=volser** parameter is mandatory.

---

### Other Commands

<table>
<thead>
<tr>
<th>QUERY</th>
<th>CBLNAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>EJECT</td>
<td></td>
</tr>
</tbody>
</table>
| EJ    | "Unconditional new page (command not printed)"

Notes

1. See **QUERY** and **EJECT** in the A-Z Reference.

---

### LISTVCAT Output Fields

1. **Standard Catalog Report**
2. **Customised Catalog Report**

#### Standard Catalog Report

For this table the assumption is made that the default report fields are those set in the CBL supplied **CBLNAME**.

<table>
<thead>
<tr>
<th>VCAT Fieldname</th>
<th>Std Pos</th>
<th>Field Heading</th>
<th>Field Format</th>
<th>Dflt Width</th>
<th>OPTION Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a (1)</td>
<td>041</td>
<td>PCNT ---- ALLOC xxxCKS ----</td>
<td><strong>nn.n</strong> + 3 alloc fields</td>
<td>32</td>
<td>PCNT*</td>
</tr>
<tr>
<td>n/a (2)</td>
<td>041</td>
<td>------ ALLOCATED xxxCKS ----</td>
<td>4 alloc fields</td>
<td>32</td>
<td>UNUSED</td>
</tr>
<tr>
<td>AVRL</td>
<td>078</td>
<td>AVRL</td>
<td>+nnnnn*</td>
<td>7</td>
<td>AVRL</td>
</tr>
<tr>
<td>BUFSP/Ixl</td>
<td>100</td>
<td>BUFSP</td>
<td>+nnnnn*</td>
<td>IXL=nn</td>
<td>7</td>
</tr>
<tr>
<td>CI/CA</td>
<td>107</td>
<td>CI/CA</td>
<td>+nnnn*</td>
<td>6</td>
<td>CI/CA</td>
</tr>
<tr>
<td>CISIZE</td>
<td>093</td>
<td>CISIZE</td>
<td>+nnnnn*</td>
<td>7</td>
<td>CI/CA</td>
</tr>
<tr>
<td>DEFINED</td>
<td>113</td>
<td>DEFINED</td>
<td>*ccyy/mm/dd</td>
<td>9</td>
<td>DEFINED</td>
</tr>
<tr>
<td>DSN (22)</td>
<td>001</td>
<td>USERCAT vvvvvv (nnnn)</td>
<td>xxxxxxxxx...</td>
<td>44</td>
<td></td>
</tr>
<tr>
<td>EXCPS</td>
<td>107</td>
<td>EXCPS</td>
<td>+nnnnK*</td>
<td>7</td>
<td>EXCPS*</td>
</tr>
<tr>
<td>FRSP</td>
<td>072</td>
<td>FRSP</td>
<td>+nn+nn*</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>KL/BLK/IMB</td>
<td>085</td>
<td>KL, RKP</td>
<td>nnn,nnn</td>
<td>+nnnnnnn</td>
<td>7</td>
</tr>
<tr>
<td>LMAX</td>
<td>078</td>
<td>LMAX</td>
<td>+nnnnnV</td>
<td>7</td>
<td>LMAX*</td>
</tr>
<tr>
<td>NRECS</td>
<td>030</td>
<td>NRECS</td>
<td>+nnnnnnnn+</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>TIMESTMP</td>
<td>112</td>
<td>TIMESTMP</td>
<td>*ccyy/mm/dd hh.mm.ss</td>
<td>20</td>
<td>TIMESTMP*</td>
</tr>
<tr>
<td>TYPE</td>
<td>024</td>
<td>TYPE</td>
<td>xxxxxxxxxx</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

Notes

1. See **LISTVCAT** in the A-Z Reference.
2. Where there is a field **OPTION** within the standard report, the default setting is indicated with an "***" in the **OPTION** required column.
3. n/a (1) - When the option **PCNT** is used, the **PCNT** column is displayed, together with three allocation columns. This is the equivalent of specifying **ALLOC3** and **PCNT** as parameters for the customised **REPORT** command.
4. n/a (2) - When the option **UNUSED** is used, four allocation columns are displayed. This is the equivalent of specifying **ALLOC4** as a parameter to the **REPORT** command.
5. **"DSN (22)"** above indicates that the standard **DSN** field is 22 bytes wide. The printing of longer names is governed by the **OVLAY/NOOVLAY** option.
### Customised Catalog Report

Fields within a customised report are chosen with the **REPORT** command and its parameters. There is no standard position for these fields as their position within the parameter list governs the report layout. All the VCAT fields in the **Standard Catalog Report** table above are also available to the **REPORT** command (the VCAT fieldname is used as the **REPORT** parameter - ignore the **OPTION Required** column and see notes after this table).

#### Summary of Syntax  Customised Catalog Report

<table>
<thead>
<tr>
<th>REPORT Parameter</th>
<th>Field Heading</th>
<th>Field Format</th>
<th>Dflt Width</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALLOC3</td>
<td>---- ALLOC xxxCKS ----</td>
<td>3 alloc fields</td>
<td>26</td>
</tr>
<tr>
<td>ALLOC4</td>
<td>---- TOTAL PRIME SEC ----</td>
<td>4 alloc fields</td>
<td>32</td>
</tr>
<tr>
<td>ALLOC5</td>
<td>ALLOC +nnnnnnnnn</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>ALLOC6</td>
<td>ALLOC +nnnnnnnnn</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>VOL1</td>
<td>VOLSER nnnnnnnn</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>VOL2</td>
<td>VOLSER DEVICE nnnnnn xxxxxxx</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>VOL3</td>
<td>SEQ VOLSER nn nnnnnnn</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>VOL4</td>
<td>SEQ VOLSER DEVCSEQ nnn nnnnnn xxx</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>VOL5</td>
<td>SEQ VOLSER DEVICE FSEQ nnn nnnnnn xxx nnn</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>VOL6</td>
<td>VOLSER nnnnnn xxx</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>

#### Notes

1. See **LISTVCAT** and **REPORT** in the A-Z Reference.

2. **ALLOC3** and **ALLOC4** form part of the standard report depending upon which (mutually exclusive) **OPTION**s are in force. **ALLOC3** forms part of the output (with the additional **PCNT** column) if the option **PCNT** is in force, whereas **ALLOC4** forms part of the output if the option **UNUSED** is in force.

3. For a Customised report, the **DSN** field has a default width of 44. If the **DSN** parameter is supplied with a numerical value to override this width and this value is less than 44, the whole data set name will be printed and all subsequent information will continue on the next line of the report (unless **SORT** has been used in which case the data set name will be truncated accordingly).
LISTVTOC Output Fields

1. Standard VTOC Report
2. Customised VTOC Report

Standard VTOC Report

<table>
<thead>
<tr>
<th>VTOC Fieldname</th>
<th>Pos</th>
<th>Field Heading</th>
<th>Field Format</th>
<th>Default Width</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALLOC</td>
<td>047</td>
<td>ALLOC</td>
<td>+nnnnnn</td>
<td>7</td>
</tr>
<tr>
<td>BLKSIZE</td>
<td>078</td>
<td>BLKSIZE</td>
<td>+nnnnnn</td>
<td>7</td>
</tr>
<tr>
<td>CISIZE (for FBA)</td>
<td>033</td>
<td>CISIZE</td>
<td>+nnnnn</td>
<td>6</td>
</tr>
<tr>
<td>CREATED</td>
<td>100</td>
<td>CREATED</td>
<td>ccyy/mm/dd</td>
<td>10</td>
</tr>
<tr>
<td>CYL/HD (for CKD)</td>
<td>023</td>
<td>CYL/HD LIMITS</td>
<td>+nnnn/nn+nnnn/nn</td>
<td>16</td>
</tr>
<tr>
<td>DSN (21/31)</td>
<td>001</td>
<td>VTOC OF VVVVV</td>
<td>xxxxxxxxxx...</td>
<td>44</td>
</tr>
<tr>
<td>EXPIRES</td>
<td>067</td>
<td>EXPIRES</td>
<td>*ccyy/mm/dd</td>
<td>11</td>
</tr>
<tr>
<td>INFO</td>
<td>112</td>
<td>INFO</td>
<td>xxxxx...</td>
<td>21</td>
</tr>
<tr>
<td>LRECL</td>
<td>093</td>
<td>LRECL</td>
<td>+nnnnn</td>
<td>6</td>
</tr>
<tr>
<td>RECFM</td>
<td>093</td>
<td>RECFM</td>
<td>xxxxx</td>
<td>5</td>
</tr>
<tr>
<td>START</td>
<td>039</td>
<td>START</td>
<td>+nnnnnnnn</td>
<td>8</td>
</tr>
<tr>
<td>TYPE</td>
<td>061</td>
<td>TYPE</td>
<td>xxxx</td>
<td><em>EXT=n</em></td>
</tr>
<tr>
<td>USED</td>
<td>054</td>
<td>USED</td>
<td>+nnnnnn</td>
<td>7</td>
</tr>
</tbody>
</table>

Notes

1. See LISTVTOC in the A-Z Reference.
2. "DSN (21/31)" indicates that the standard DSN field is 21 bytes wide (For CKD) and 31 bytes wide (For FBA). The printing of any longer names is governed by the setting of the OVLAY/NOOVLAY option.

Customised VTOC Report

Fields within a customised report are chosen with the REPORT command and its parameters. There is no standard position for these fields as their position within the parameter list governs the report layout. All the VTOC fields in the Standard VTOC Report table above are also available to the REPORT command (the VTOC fieldname is used as the REPORT parameter).

<table>
<thead>
<tr>
<th>REPORT Parameter</th>
<th>Field Heading</th>
<th>Field Format</th>
<th>Dflt Width</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCESSED</td>
<td>ACCESSED</td>
<td>ccyy/mm/dd</td>
<td>10</td>
</tr>
<tr>
<td>EXPIRES</td>
<td>EXPIRES</td>
<td>ccyy/mm/dd*</td>
<td>11</td>
</tr>
<tr>
<td>VOLUME</td>
<td>VOLUME</td>
<td>volser</td>
<td>6</td>
</tr>
</tbody>
</table>

Notes

1. See LISTVTOC and REPORT in the A-Z Reference.
2. For a Customised report, the DSN field has a default width of 44. If the DSN parameter is supplied with a numerical value to override this width and this value is less than 44, the whole data set name will be printed and all subsequent information will continue on the next line of the report (unless SORT has been used in which case the data set name will be truncated accordingly).
# Abbreviations and Synonyms

The following abbreviations are supported. Abbreviated and full length keywords may be mixed at will.

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Length Keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>DEV=ALL</td>
</tr>
<tr>
<td>ALL</td>
<td>REF=ALL</td>
</tr>
<tr>
<td>AVRECL</td>
<td>AVLRECL</td>
</tr>
<tr>
<td>D</td>
<td>DEL</td>
</tr>
<tr>
<td>DD</td>
<td>DDNAME</td>
</tr>
<tr>
<td>DEF</td>
<td>DEFINE</td>
</tr>
<tr>
<td>EJ</td>
<td>EJECT</td>
</tr>
<tr>
<td>EXTNUM</td>
<td>EXTNO</td>
</tr>
<tr>
<td>L</td>
<td>LSTVTOC</td>
</tr>
<tr>
<td>LC</td>
<td>LISTVCAT</td>
</tr>
<tr>
<td>LIST</td>
<td>LISTVTOC</td>
</tr>
<tr>
<td>LISTC</td>
<td>LISTVCAT</td>
</tr>
<tr>
<td>LISTCAT</td>
<td>LISTVCAT</td>
</tr>
<tr>
<td>LISTL</td>
<td>LISTLABL</td>
</tr>
<tr>
<td>LISTV</td>
<td>LISTVTOC</td>
</tr>
<tr>
<td>LL</td>
<td>LISTLABL</td>
</tr>
<tr>
<td>LV</td>
<td>LSTVTOC</td>
</tr>
<tr>
<td>MAXRECL</td>
<td>MAXLRECL</td>
</tr>
<tr>
<td>OPT</td>
<td>OPTIONS</td>
</tr>
<tr>
<td>OPTION</td>
<td>OPTIONS</td>
</tr>
<tr>
<td>OWNER</td>
<td>OWN</td>
</tr>
<tr>
<td>PD</td>
<td>PAGEDEPTH</td>
</tr>
<tr>
<td>PW</td>
<td>PAGEWIDTH</td>
</tr>
<tr>
<td>Q</td>
<td>QUERY</td>
</tr>
<tr>
<td>SORTA</td>
<td>SORT</td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>TIMESTAMP</td>
</tr>
<tr>
<td>TYPE</td>
<td>TYPE</td>
</tr>
</tbody>
</table>

**Notes**

If the synonym **ALL** is used, its meaning is dependent upon the command it follows (i.e. DEV=ALL for LISTVTOC, or REF=ALL for LISTVCAT).
The following table contains all CBLVCAT's Commands and Parameters (Commands are highlighted).

<table>
<thead>
<tr>
<th>! (Separator)</th>
<th>CBLVCSW8</th>
<th>HEAD</th>
<th>NOEXPD</th>
<th>SEV</th>
</tr>
</thead>
<tbody>
<tr>
<td>* (Comment)</td>
<td>CBLVCSW9</td>
<td>HIALLRBA</td>
<td>NOFREE</td>
<td>SEVL</td>
</tr>
<tr>
<td>ACCESSED</td>
<td>CI/CA</td>
<td>HIBLK</td>
<td>NOGDGSN</td>
<td>SHR</td>
</tr>
<tr>
<td>ALIAS</td>
<td>CIMAX</td>
<td>HICISZ</td>
<td>NOGDGRPT</td>
<td>SMS</td>
</tr>
<tr>
<td>ALLFILES</td>
<td>CIMIN</td>
<td>HICYL</td>
<td>NOINDEX</td>
<td>SMUSD</td>
</tr>
<tr>
<td>ALLOC</td>
<td>CISIZE</td>
<td>HIDATE</td>
<td>NOISC</td>
<td>SMMS</td>
</tr>
<tr>
<td>ALLOC3</td>
<td>CLASS</td>
<td>HIEXP</td>
<td>NOMOUNT</td>
<td>SMSS</td>
</tr>
<tr>
<td>ALLOC4</td>
<td>COMPONENT</td>
<td>HITRK</td>
<td>NOOVLAY</td>
<td>SORT</td>
</tr>
<tr>
<td>ALLOCP</td>
<td>COMPRESSED</td>
<td>HIUSERBA</td>
<td>NOPCAP</td>
<td>SORTD</td>
</tr>
<tr>
<td>ALLOCS</td>
<td>CREATED</td>
<td>IGN</td>
<td>NOPERM</td>
<td>SPANNED</td>
</tr>
<tr>
<td>ALLOCT</td>
<td>CYL/HDL</td>
<td>IMB</td>
<td>NOPJCL</td>
<td>SPLIT</td>
</tr>
<tr>
<td>ALLOCU</td>
<td>CYLMAX</td>
<td>INDEX</td>
<td>NOPSEV</td>
<td>SPLITCA</td>
</tr>
<tr>
<td>ASSOC</td>
<td>DDNAME</td>
<td>INFO</td>
<td>NOVOL</td>
<td>SPLITCSI</td>
</tr>
<tr>
<td>AVLRECL</td>
<td>DEFINE</td>
<td>ISC</td>
<td>NRECS</td>
<td>SPLIT</td>
</tr>
<tr>
<td>AVRL</td>
<td>DEFINED</td>
<td>IXL</td>
<td>NSEC</td>
<td>START</td>
</tr>
<tr>
<td>BLKSIZE</td>
<td>DEL</td>
<td>KEY</td>
<td>OPTIONS</td>
<td>STOPAFT</td>
</tr>
<tr>
<td>BUFSP</td>
<td>DEV</td>
<td>KL</td>
<td>OVLAY</td>
<td>STRIPED</td>
</tr>
<tr>
<td>BUFSP/IXL</td>
<td>DSN</td>
<td>KL/BLK/IMB</td>
<td>OWN</td>
<td>SUBSET</td>
</tr>
<tr>
<td>CAT</td>
<td>EJECT</td>
<td>LINESPACE</td>
<td>PAGEDEPTH</td>
<td>SUMMARY</td>
</tr>
<tr>
<td>CATALOG</td>
<td>ENTRY</td>
<td>LIST</td>
<td>PAGEWIDTH</td>
<td>SYS</td>
</tr>
<tr>
<td>CBLCLINE</td>
<td>EXCPS</td>
<td>LISTLABEL</td>
<td>PASS</td>
<td>TIMESTAMP</td>
</tr>
<tr>
<td>CBLVCALE</td>
<td>EXP</td>
<td>LISTVCAT</td>
<td>PCNT</td>
<td>TOTALLOC</td>
</tr>
<tr>
<td>CBLVCALW</td>
<td>EXPD</td>
<td>LISTVTOC</td>
<td>PERM</td>
<td>TOTALS</td>
</tr>
<tr>
<td>CBLVCEXT</td>
<td>EXPERES</td>
<td>LMAX</td>
<td>PHYREC</td>
<td>TUNE</td>
</tr>
<tr>
<td>CBLVCFN</td>
<td>EXT</td>
<td>LOBLK</td>
<td>PRTCAP</td>
<td>TYPE</td>
</tr>
<tr>
<td>CBLVCONL</td>
<td>EXT-ADDR</td>
<td>LOCISZ</td>
<td>PRTJCL</td>
<td>UNALLOC</td>
</tr>
<tr>
<td>CBLVCCONS</td>
<td>EXTENDED</td>
<td>LOCYL</td>
<td>PRTSEV</td>
<td>UNIT</td>
</tr>
<tr>
<td>CBLVCONT</td>
<td>EXTNO</td>
<td>LODATE</td>
<td>QUERY</td>
<td>UNUSED</td>
</tr>
<tr>
<td>CBLVCPCF</td>
<td>FAIL</td>
<td>LOEXP</td>
<td>RAW=fname</td>
<td>USED</td>
</tr>
<tr>
<td>CBLVCPCCT</td>
<td>FREE</td>
<td>LOTTREK</td>
<td>RECDEL</td>
<td>VCAT</td>
</tr>
<tr>
<td>CBLVCPCV</td>
<td>FREEBYTES</td>
<td>LRECL</td>
<td>RECFM</td>
<td>VOL</td>
</tr>
<tr>
<td>CBLVCRCM</td>
<td>FREETAB</td>
<td>MAXLRECL</td>
<td>RECINP</td>
<td>VOL1</td>
</tr>
<tr>
<td>CBLVCSCA</td>
<td>FRSP</td>
<td>MERGE</td>
<td>RECONS</td>
<td>VOL2</td>
</tr>
<tr>
<td>CBLVCSCI</td>
<td>FRSPCA</td>
<td>MOD</td>
<td>RECORDS</td>
<td>VOL3</td>
</tr>
<tr>
<td>CBLVCSPA</td>
<td>FRSPCI</td>
<td>MOUNT</td>
<td>RECSSTATS</td>
<td>VOL4</td>
</tr>
<tr>
<td>CBLVCSSW1</td>
<td>GDGDSN</td>
<td>NEWDSN</td>
<td>RECUPD</td>
<td>VOL5</td>
</tr>
<tr>
<td>CBLVCSSW2</td>
<td>GDGRPT</td>
<td>NEWOWN</td>
<td>REF</td>
<td>VOLINFO</td>
</tr>
<tr>
<td>CBLVCSSW3</td>
<td>GGEN</td>
<td>NEWVOL</td>
<td>REPORT</td>
<td>VOLUME</td>
</tr>
<tr>
<td>CBLVCSSW4</td>
<td>GMAX</td>
<td>NOALIAS</td>
<td>RKP</td>
<td>VTOC</td>
</tr>
<tr>
<td>CBLVCSSW5</td>
<td>GROWTH</td>
<td>NOASSOC</td>
<td>S/C</td>
<td>VVDS</td>
</tr>
<tr>
<td>CBLVCSSW6</td>
<td>GVER</td>
<td>NODEDEFAULT</td>
<td>SECALLOC</td>
<td>XYVAM</td>
</tr>
<tr>
<td>CBLVCSSW7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**! (Separator Character)**

```
LISTVTOC  SYS=022  !LISTVTOC SYS=023  !LISTVTOC SYS=024
LISTVCAT  DD=VSESPUC  * VSE User Catalog  !LISTVCAT DD=CBLMCT
```

The control statement separator character is defined in CBLNAME which is distributed with "!" (exclamation mark - EBCDIC X'5A') as the default.

This allows the user to code multiple CBLVCAT statements on a single SYSIN/SYSIPT input record. It defines logical end of the CBLVCAT control statement, is not considered to be part of the record and not printed in the SYSPRINT/SYSLST output.
REPORT VCAT  DSN 33  ALLOC3 NSEC                  * Catalog Report Fields.
REPORT VTOC  DSN  TYPE  CREATED ACCESSED EXPIRES  * Catalog Report Fields.

** Generate Lists **
LISTVTOC  VOL=CBLI03                  * User Volume CBLI03.
LISTVCAT  KEY=CBL.DIST              * All cataloged data sets with prefix "CBL.DIST".

An asterisk (X'5C' EBCDIC) in position 1, or in any position which is preceded by a blank and is not enclosed in quotes, defines the start of comment data. The comment data ends at the end of SYSIN/SYSIPT input record or at a command separation character ("!" by default).

Comment data is ignored by CBLVCAT for syntax purposes.

** ACCESSSED (nn) **

REPORT VTOC  DSN  CREATED ACCESSED

For MVS systems only, ACCESSSED displays the date the file was last accessed.

The column width is 10, the format is 'ccyy/mm/dd' and the heading is ACCESSSED.

Notes

If CBLVCSW9=X'01' is set on (CBLNAME option V2digitYear=Yes) then the format of the ACCESSSED field is 'yy/mm/dd'.

** ALIAS **

OPTION ALIAS                  * Override CBLVCSW1=X'08'
LC  DD=CBLI02

For MVS systems only. If NOALIAS is the current installation default, ALIAS can be used to display the ALIAS items as separate entries in the report (They will still appear as a reference for the appropriate USERCAT). ALIAS and NOALIAS are mutually exclusive options.

In the CBL supplied version of CBLNAME, ALIAS is the default (CBLVCSW1 bit X'08' is on). NOALIAS can be made the default by changing this setting.

Example: LISTVCAT ALIAS Output (CBL Ref: vmxalia)
**ALLFILES=YES**

For VSE systems only, **ALLFILES=YES** selects all files on the volume for deletion (including any unexpired datasets). **ALLFILES=YES** and **DSN=xxx** are mutually exclusive.

Select the device using the **DEV** and/or **SYS** parameters and check the volume via the **VOL** parameter and, if required, the **OWN** parameter. The **VOL** parameter is **mandatory**, as a safety mechanism to ensure that the correct volume is mounted.

---

**ALLOC (nn)**

Display the space allocated to a file or free space extent.

The column width is 7, the format is +nnnnnn (tracks for CKD or blocks for FBA) and the heading is **ALLOC**. A total is provided for this column (ALLOC also forms part of the standard report).

---

**ALLOCP (nn)**

Display the primary allocation (see **ALLOCT** for an example).

The column width is 8 and the format is either +nnnnnnn (tracks for CKD or blocks for FBA) or C=nnn (cylinders for CKD). The heading is **ALLOC PRIME** and a total is provided (This column is also displayed within the allocated field of the standard report).

**Notes**

If a file extends onto a **CANDIDATE** volume, a second primary allocation is acquired.

A primary allocation can have multiple extents (see **NSEC** to display number of extents).

---

**ALLOCS (nn)**

Display the secondary allocation.

The column width is 10 and the format is either +nnnnnn*nn (tracks for CKD or blocks for FBA) or C=nnn (cylinders for CKD only). The heading is **ALLOC SEC** and a total is provided for this column (This column is also part of the allocated field of the standard report).
Example: ALLOCS and NSEC fields (CBL Ref: vmxnsec)

Notes

If 0* is displayed (see example for ALLOCS), it indicates that the file has been defined with no secondary allocation specified (This may be appropriate if the file is static). The asterisk will not appear if GROWTH=0 is displayed at tuning time.

All extents other than the first extent of the first Primary Allocation are Secondary extents. Therefore, if a primary allocation consists of 3 extents, the number of secondary extents is 2 plus the number of extents reserved by any secondary allocation plus the number of extents reserved by any further primary allocations (See note in ALLOCP above).

In the above example, C=25*6 does not necessarily mean 6 secondary allocations of 25 cylinders each.

ICF and VSAM support up to 123 extents, except for VSAM cataloged REUSE or UNIQUE files which are restricted to 16 extents per volume.

One asterisk (nnn*) indicates that the number of secondary extents has reached the warning threshold limit. The value after the asterisk shows how many times secondary extents have been acquired (The warning threshold limit is 4 by default, but can be set to another value in CBLNAME using the CBLVCALW field). SEV 2-19 would also be activated.

Three asterisks (nnn***) indicate that the number of secondary extents has reached the error threshold limit (This is 10 by default, but can be set to another value in CBLNAME using the CBLVCALE SEV 3-19 would also be activated, however, the number of extents is not displayed. The NSEC field would be needed if a count of the secondary extents was required.

ALLOCT (nn) (REPORT VCAT)

Display the current total space allocation of a component.

The column width is 8 and the format is either +nnnnnnn (tracks for CKD or blocks for FBA) or C=nnn (cylinders for CKD only). The heading is ALLOC TOTAL and a total is provided for this column (This column is also part of the allocated field of the standard report).
is therefore possible to see if there is enough space left for all the files to be open at once (however, it is unlikely that this would be

The total blocks/tracks these files will occupy is reported separately at the end of the report (see Example 2 in Guide to LIST Output). It

Files shown as TEMP are temporary reusable files (VSE only), which means that they have been defined as WORK files. These files are not allocated until they are opened for output, at which time the required space is taken from unused remaining space. The total blocks/tracks these files will occupy is reported separately at the end of the report (see Example 2 in Guide to LIST Output). It is therefore possible to see if there is enough space left for all the files to be open at once (however, it is unlikely that this would be required).

Example: LISTVCAT allocation totals (CBL Ref: vmxallt)

Notes

This column is not always the sum of the PRIME and SEC columns. It is either the same as the PRIME value (if no secondary allocations have occurred) or the PRIME value plus a multiple of the SEC value (if secondary allocations have occurred).

However, each time a file extends onto a candidate volume, an additional primary allocation is acquired. In this case the total would simply be a multiple of the PRIME column (or the total could be a multiple of both PRIME and SEC columns if secondary allocations had also been acquired). See VOLINFO for the display of volume information.

Files shown as TEMP are temporary reusable files (VSE only), which means that they have been defined as WORK files. These files are not allocated until they are opened for output, at which time the required space is taken from unused remaining space. The total blocks/tracks these files will occupy is reported separately at the end of the report (see Example 2 in Guide to LIST Output). It is therefore possible to see if there is enough space left for all the files to be open at once (however, it is unlikely that this would be required).

Notes

This column is not always the sum of the PRIME and SEC columns. It is either the same as the PRIME value (if no secondary allocations have occurred) or the PRIME value plus a multiple of the SEC value (if secondary allocations have occurred).

However, each time a file extends onto a candidate volume, an additional primary allocation is acquired. In this case the total would simply be a multiple of the PRIME column (or the total could be a multiple of both PRIME and SEC columns if secondary allocations had also been acquired). See VOLINFO for the display of volume information.

Files shown as TEMP are temporary reusable files (VSE only), which means that they have been defined as WORK files. These files are not allocated until they are opened for output, at which time the required space is taken from unused remaining space. The total blocks/tracks these files will occupy is reported separately at the end of the report (see Example 2 in Guide to LIST Output). It is therefore possible to see if there is enough space left for all the files to be open at once (however, it is unlikely that this would be required).

Notes

This column is not always the sum of the PRIME and SEC columns. It is either the same as the PRIME value (if no secondary allocations have occurred) or the PRIME value plus a multiple of the SEC value (if secondary allocations have occurred).

However, each time a file extends onto a candidate volume, an additional primary allocation is acquired. In this case the total would simply be a multiple of the PRIME column (or the total could be a multiple of both PRIME and SEC columns if secondary allocations had also been acquired). See VOLINFO for the display of volume information.

Files shown as TEMP are temporary reusable files (VSE only), which means that they have been defined as WORK files. These files are not allocated until they are opened for output, at which time the required space is taken from unused remaining space. The total blocks/tracks these files will occupy is reported separately at the end of the report (see Example 2 in Guide to LIST Output). It is therefore possible to see if there is enough space left for all the files to be open at once (however, it is unlikely that this would be required).
Example: LISTVCAT sealloc warnings (CBL Ref: vmxsecw)

Notes

One asterisk (**nnn*) indicates that the number of secondary extents has reached the warning threshold limit. The value after the asterisk shows how many times secondary extents have been acquired (The warning threshold limit is 4 by default, but can be set to another value in the CBLNAME module using the CBLVCALW field). SEV 2-19 is also activated.

Three asterisks (**nnn***) indicate that the number of secondary extents has reached the error threshold limit (This is 10 by default, but can be set to another value in the CBLNAME module using the CBLVCALE field). SEV 3-19 is also activated.

For the standard report only, when this limit is reached a warning line is also printed indicating the number of secondary extents (see above). If this information is required for a customised report the NSEC field should be used. Actual extent information can also be shown depending upon the setting of the CBLNAME module using the CBLVCXT field).

Zero flagged with an asterisk on the right (**0*) is a secondary allocation warning message, meaning that the file has been defined with no secondary allocation allowed (This may be appropriate if the file is static). The asterisk will not appear if GROWTH=0 is specified at tuning time.

**ALLOC4 (nn)**

Display the 4 columns TOTAL, UNUSED, PRIME and SEC.

The column width is 32, comprising the ALLOC, ALLOCU, ALLOCP, ALLOCS columns (described earlier) and the heading is ---- ALLOCATED xxxxxS ----, where “xxxxx” is “TRACK” (CKD) or “BLOCK” (FBA). The columns which comprise ALLOC4 also form part of the standard report if option UNUSED / PCNT is set to UNUSED).

See Notes in ALLOC3 above.

**ASSOC**

If NOASSOC is the current installation default, ASSOC can be used to display file association information on the Catalog report (ASSOC and NOASSOC are mutually exclusive options).

In the CBL supplied version of CBLNAME, ASSOC is the default (CBLVCSW1 bit X’80’ is on). NOASSOC can be made default by changing this switch.

Note

For VSAM (non ICF) catalogs, a SUBSET using KEY and/or IGN, will process much more quickly if NOASSOC is specified (otherwise CBLVCAT still processes all the files). ICF catalogs have ASSOC information included within the cluster ‘sphere’ record, so ASSOC/NOASSOC has little impact on performance.

**ASSOC (nn)**

Display the name of the user catalog (or non-VSAM entry) for which an ALIAS has been defined.

The column width is 44 and the column heading is ASSOC.
AVLRECL=nnn/KEEP

Sets the average record length to be used for a tuning run.

It is provided as an override to CBLVCAT's estimated value of the average record length (AVLRECL=KEEP, uses the defined average record length, therefore, the user does not need to explicitly define an AVLRECL=nnn for each file if no change to the average record length is required).

File Modelling

AVLRECL is useful when modelling a new file on the attributes of an existing file. It allows a tuning run to provide recommendations based on the existing file, but tailored to the average record length expected for the new file.

File Fragmentation

CI or CA splits occurring at record insertion time may result in CBLVCAT being unable to judge the extent of Freespace distribution. This can lead to a poor estimation of average record length (see Average Record Length Estimation in Tuning Considerations).

AVLRECL and Maximum Record Length

The specification of AVLRECL can also affect CBLVCAT's recommendation of Maximum Record Length (See LMAX). If the specified AVLRECL is less than the current maximum record length, then the maximum record length value is retained. However, if the specified average record length is greater than the current maximum record length then the maximum record length value is set equal to the specified AVLRECL value.

AVLRECL and MAXLRECL

Specifying AVLRECL combined with MAXLRECL (for basing tuning recommendations on larger average and maximum record lengths) can result in CBLVCAT also recommending a larger CI size.

Alternate Index Datasets which were defined with default values for MAXIMUM and AVERAGE record size (32600 and 4086 respectively), can cause CBLVCAT to over allocate capacity in its tuning recommendations (A WARN 14 message is provided). Supplying more accurate values via the MAXLRECL and AVLRECL parameters will overcome this problem.

AVRL

OPT  CI/CA  AVRL  ASSOC

If LMAX (maximum record length) is the current installation default, AVRL can be used to display the defined average record length. AVRL and LMAX are mutually exclusive OPTIONS.

In the CBL supplied version of CBLNAME, LMAX is the default (CBLVCSW3 bit X'10' is off), however AVRL may be made the default by changing this switch.

Note

It is possible for the actual average record length of the loaded data to be different from the defined average record length (See severity message SEV 2-31).

AVRL (nn)

REPORT VCAT DSN  TYPE  NRECS  LMAX  AVRL

Display the column containing the defined average record length.

The column width is 7, the format is +nnnnn* and the heading is AVRL (AVRL forms part of the standard Catalog report if the LMAX / AVRL option is set to AVRL).

See Note above.
### BLKSIZE (nn)

**REPORT VCAT**  
**DSN 30**  
**TYPE BLKSIZE**

For **VSE** systems only, display the blocksize for SAM files which reside in a VSAM space.  
The column width is 7, the format is +nnnnn* and the heading is **BLKSIZE**  
(VSAM SAM BLKSIZE details are also available in a standard report, via the **KL/BLK/IMB** combined column).

### BLKSIZE (nn)

**REPORT VTOC**  
**DSN**  
**TYPE SORTD BLKSIZE LRECL**

Display the blocksize for **MVS** files.  
The column width is 7, the format is +nnnnnn and the heading is **BLKSIZE** (**BLKSIZE** also forms part of the standard report).

VSE SAM does not automatically record a file's geometry (RECFM, LRECL and BLKSIZE) in disk VTOCs. However, it is common for VSE applications that perform I/O to include this information in the DTF control block and so write the file's geometry to its Format 1 record in the VTOC. **CBLVCAT** reports the blocksize value, if present, otherwise blanks are displayed in the **BLKSIZE** column.

### BUFSP (nn)

**REPORT VCAT**  
**DSN 30**  
**TYPE SORTD BUFSP**

Display the Bufferspace supplied to VSAM when the file was defined.  
The column width is 7, the format is +nnnnn* and the heading is **BUFSP** (Bufferspace details are also available as part of the standard report via the **BUFSP/IXL** combined column).

**Notes**

BUFSP can be *increased* at application run time via JCL overrides (However it cannot be decreased). The optimum buffer space for VSAM to use, depends on whether the file will be processed sequentially or randomly and also on the required number of index levels (see Bufferspace in Additional VSAM Information).

An asterisk appended to the right of the **BUFSP** value (nnnnn*) indicates that the allocated bufferspace is too small to accommodate 2 data CIs plus 1 index CI, for each level of index plus 1 (CBLVCAT recommends 1 index buffer more than the levels of index, in case CI/CA splits increase the number of index levels).

### BUFSP/IXL (nn)

**REPORT VCAT**  
**DSN 30**  
**TYPE SORTD BUFSP/IXL CISIZE EXCPS**

A combined field which displays the **Bufferspace** for a data component and the **Number of Index Levels** for an index component.  
The column width is 7, the format is +nnnnn* or IXL=nn and the column heading is **BUFSP/IXL** (**BUFSP/IXL** also forms part of the standard report).
Example: LISTVCAT BUFS/P and IXL field (CBL Ref: vmxbufx)

**Notes**

**SPANNED** shows that the file has been defined with the **SPANNED** attribute, which allows records to span control intervals. If the maximum record length is only slightly greater than the CI size, CBLVCAT tuning will recommend a larger CI size and **NONSPANNED**, as **SPANNED** processing is costly on machine resources.

The two columns are available separately via the **BUFS** and **IXL** parameters.

See **BUFS** for a description of the bufferspace warnings.

**CAT=xxx.xx**

For **VSE** systems only, **CAT** may be specified together with the full catalog data set name to select the required Catalog as an alternative to **DDNAME=label**.

**Notes**

CBLVCAT will dynamically allocate the arbitrary, temporary label, **CBLVL54**, to the specified catalog data set name in order to open and read the catalog. (L54 indicates that the length of the label information is 54 bytes).

This method means that a **DLBL statement** is not required as it one is dynamically allocated to the Catalog name.

**CATALOG (nn)**

**Display the dataset name of the catalog in which the file is defined.**

The column width is 44 and the heading is **CATALOG**.

**Note**

This field can be especially useful when using **MERGE** to produce reports containing information from more than one catalog (see example 16 in Guide to List Output).

**CBLCLINE=nn**

**A run time override of the page depth.**
The default can be set using the **CBLCLINE** (CBL Common LINEs) field within **CBLNAME**, which is supplied containing the value **X'00'** (58 lines for MVS and the SYSLST system default for VSE).

**Note**

Specifying **CBLCLINE=60** is the same as supplying **CBLCLINE=X'3C'**, and is also the same as **PAGEDEPTH=60**, or **PD=60**.

Setting a very high **PAGEDEPTH**, e.g. 32767 (32K-1), causes no page throws and consequently only one set of headings. This can be a useful technique when **post-processing** reports.

### CBLVCALE=nn

**LISTVCAT OPTION**

<table>
<thead>
<tr>
<th>OPTION</th>
<th>CBLVCALW=10</th>
<th>CBLVCALE=20</th>
</tr>
</thead>
</table>

A run time override of the error threshold for secondary extents.

The default can be set using the **CBLVCALE** (CBL VCat ALloc Error) field within **CBLNAME**, which is supplied containing the value **X'0A'** (10).

**Note**

Reaching or exceeding this threshold will trigger a **SEV 3-nn** severity message, indicating that the file should be tuned as soon as possible. Three asterisks (***** **) will also be appended to the secondary allocation value (see **ALLOCS**).

The **CBLVCALE** value **MUST** be greater than that of **CBLVCALW** to have any affect.

### CBLVCALW=nn

**LISTVCAT OPTION**

<table>
<thead>
<tr>
<th>OPTION</th>
<th>CBLVCALW=10</th>
<th>CBLVCALE=20</th>
</tr>
</thead>
</table>

A run time override of the warning threshold for secondary extents.

The default can be set using the **CBLVCALW** (CBL VCat ALloc Warning) field within **CBLNAME**, which is supplied containing the value **X'04'** (4).

**Note**

Reaching or exceeding this threshold will trigger a **SEV 2-nn** severity message, indicating that the file should be tuned when next due for re-loading. An asterisk followed by the number of secondary extents (**nn**) will also be appended to the secondary allocation value (see **ALLOCS**).

### CBLVCEXT=nn

**LISTVCAT OPTION**

<table>
<thead>
<tr>
<th>OPTION</th>
<th>CBLVCEXT=4 NOASSOC</th>
</tr>
</thead>
</table>

A run time override for the number of physical extents you wish to display for a file.

The default can be set using the **CBLVCEXT** (CBL VCat EXTents) field in **CBLNAME**, which is supplied containing the value **X'00'** (0).
Example: LISTVCAT with CBLVCEXT (CBL Ref: vmxextc)

Notes

This override can be useful, when used in conjunction with the LOBLK/ LOCYL/ LOTRK and HIBLK/ HICYL/ HITRK subsets, for tracking down files whose extents lie within a particular disk area.

As can be seen in the above example, once the CBLVCEXT limit is reached, the first extent for each subsequent volume is still displayed.

Care must be taken with large values for this parameter as a report line is printed for each extent, which can result in a large output print file.

---

CBLVCFN=xxx

(LISTVCAT OPTION)

A run time override for the default VSAM Catalog Name.

The default is used if the DDNAME parameter is not supplied on a LISTVCAT operation. It can be set using the CBLVCFN (CBL VCat File Name) field in CBLNAME, which is supplied containing hex zeros (giving a default of IJSYSUC).

---

CBLVCONL=X'xx'

(LISTVCAT OPTION)

A run time override for the on-line string indicator, which, if present in a file name, denotes an on-line file (Used for tuning).

The default can be set using the CBLVCONL (CBL VCat ONLINE) field in CBLNAME, which is supplied containing X'00' (not set). Byte CBLVCONL is mapped as follows:

| X'80' | select string ONLINE |
| X'40' | select string CICS   |
| X'20' | select string IMS    |
| X'10' | select string DLI    |
| X'08' | select string DL1    |
| X'06' | reserved (must be off) |
| X'01' | select string within CBLVCONS |

Notes

Not all occurrences of the string enable on-line mode. The filename can be considered to be comprised of one or more "segments" or "qualifiers", each separated by a period ("."). Only strings matching complete segments will activate "online" tuning.

When a file is encountered which contains the relevant string, tuning recommendations are given for that file, based on the setting of the CBLVCONL switch.
CBLVCONS=xxxx  (LISTVCAT OPTION)

| OPTION  | CBLVCONL=X'01' | CBLVCONS=ONLTST | CBLVCONT=512 |

A run time override, used in conjunction with CBLVCONL and CBLVCONT, to specify a user-defined character string, which, if present in a file name, denotes an on-line file (Used for tuning).

CBLVCONS can be 1 to 8 bytes in length, must represent a complete segment within the dataset name and is active if CBLVCONL bit X'01' is set on (As supplied, the default setting is hex zeros).

CBLVCONT=nnnn  (LISTVCAT OPTION)

| OPTION  | CBLVCONL=X'01' | CBLVCONS=ONLTST | CBLVCONT=512 |

A run time override for the maximum CI size for on-line files (Used for tuning).

The default can be set using the CBLVCONT, to (CBL VCat ONline Threshold) field in CBLNAME, which is supplied containing hex zeros (giving a default of 4096).

CBLVCPCF=nn  (LISTVCAT OPTION)

| LC  | DD=CBLV01 | CBLVCPCF=75 |

A run time override for the file full percentage threshold.

The default can be set using the CBLVCPCF (CBL VCat PCnt Files) field in CBLNAME, which is supplied containing X'55' (85%).

CBLVCPCT=nn  (LISTVTOC OPTION)

| LV  | VOL=CBLT22 | CBLVCPCT=90 |

A run time override for the tracks full percent threshold (for use in conjunction with the FREETAB option).

The default can be set using the CBLVCPCT (CBL VCat PCnt Tracks) field in CBLNAME, which is supplied containing X'55' (85%).

Example: LISTVTOC VOL PCNTAGE FULL (CBL Ref: vmxpcfc)

Notes

Expired (EXPD) files are counted as tracks used unless the option (NOEXPD) is set.

LISTVTOC with FREETAB reports the Percentage tracks used and also checks this value against CBLVCPCT. If this threshold is reached, the percentage value is flagged with asterisks (** nn% **) and a warning message is produced (see above).
CBLVCPCV=nn  (LISTVCAT OPTION)

LC   DD=CBLV01  CBLVCPCV=75

A run time override for the volume full percent threshold.
The default is set using the CBLVCPCV (CBL VCat PCnt Volumes) field in CBLNAME, which is supplied containing 'X'55' (85%).

CBLVCRCM=nn  (DEL,LISTVCAT,LISTVTOC,MOD OPTION)

OPTION   CBLVCRCM=16   * Suppress RC=1-15

A run time override for the Minimum return code.
The default is set using the CBLVCRCM (CBL VCat Return Code Minimum) field in CBLNAME, which is supplied containing 'X'00' (0).

Notes
Any Return Code that is below this minimum is suppressed and replaced with zero. This zero Return Code is passed to the operating system but the original value is reported in the SYSLST/SYSPRINT file.

See "Return Codes" for more information about Return Code values and the different effects for levels of VSE operating systems and CBLVCSW4 'X'04' in "CBLNAME" regarding the interaction with OEM software that does not recognise OPERATING SYSTEM Return Codes.

CBLVCSCA=nn  (LISTVCAT OPTION)

LC   DD=CBLI11   CBLVCSCI=6   CBLVCSCA=2

A run time override for the threshold value for CA Splits.
The default can be set using the CBLVCSCA (CBL VCat Splits CA) field in CBLNAME, which is supplied containing 'X'01' (1).

Note
For a Standard report, when this value is exceeded, the warning appears under the TIMESTMP/DEFINED column. For a customised report use SPLITS or SPLITCA.

CBLVCSCI=nn  (LISTVCAT OPTION)

LC   DD=CBLI11   CBLVCSCI=6   CBLVCSCA=2

A run time override for the threshold value for CI Splits.
The default can be set using the CBLVCSCI (CBL VCat Splits CI) field in CBLNAME, which is supplied containing 'X'01' (1).

Note
For a Standard report, when this value is exceeded, the warning appears under the TIMESTMP/DEFINED column. For a customised report use SPLITS or SPLITCI.

CBLVCSPA=n  (LISTVCAT,LISTVTOC OPTION)

LC   DD=CBLI11   CBLVCSPA=1   * Override LINESPACE=2

A run time override for the number of lines between files on a report.
The default can be set using the CBLVCSPA (CBL VCat SPacing) field in CBLNAME, which is supplied containing 'X'00' (gives a default spacing of 2).
For a Standard report, when this value is exceeded, the warning appears under the Specifying CBLVCSPA=2 is the same as CBLVCSPA=X'02' or LINESPACE=2.

CBLVCSW1=X'xx'

A run time override for the CBLVCSW1 print options.

The default can be set using the CBLVCSW1 (CBL VCat SWitch 1) field in CBLNAME, which is supplied containing X'FF'. Byte CBLVCSW1 is mapped as follows:

<table>
<thead>
<tr>
<th>X'80'</th>
<th>on = ASSOC option</th>
<th>off = NOASSOC option</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'40'</td>
<td>on = print --IMB-- eyecatcher (See IMB)</td>
<td></td>
</tr>
<tr>
<td>X'20'</td>
<td>on = print --REP-- eyecatcher (See IMB)</td>
<td></td>
</tr>
<tr>
<td>X'10'</td>
<td>on = VOLINFO option</td>
<td>off = NOVOL option</td>
</tr>
<tr>
<td>X'08'</td>
<td>on = (MVS) ALIAS option</td>
<td>off = (MVS) NOALIAS option</td>
</tr>
<tr>
<td>X'08'</td>
<td>on = (VSE) SHR option</td>
<td>off = (VSE) S/C option</td>
</tr>
<tr>
<td>X'04'</td>
<td>on = PCNT option</td>
<td>off = UNUSED option</td>
</tr>
<tr>
<td>X'02'</td>
<td>on = TIMESTMP option</td>
<td>off = DEFINED option</td>
</tr>
<tr>
<td>X'01'</td>
<td>on = EXCPS option</td>
<td>off = CI/CA option</td>
</tr>
</tbody>
</table>

The above options are fully documented under their option name.

CBLVCSW2=X'xx'

A run time override for the CBLVCSW2 print options.

The default can be set using the CBLVCSW2 (CBL VCat SWitch 2) field in CBLNAME, which is supplied containing X'00'. Byte CBLVCSW2 is mapped as follows:

<table>
<thead>
<tr>
<th>X'C0'</th>
<th>Reserved (should be set off)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'20'</td>
<td>on = inhibit print of Volume Serial Numbers</td>
</tr>
<tr>
<td>X'10'</td>
<td>on = FREETAB or FREE (see X'08')</td>
</tr>
<tr>
<td>X'08'</td>
<td>on = FREETAB option (and X'10' on)</td>
</tr>
<tr>
<td>X'04'</td>
<td>on = NOEXP option</td>
</tr>
<tr>
<td>X'02'</td>
<td>on = NOPERM option</td>
</tr>
<tr>
<td>X'01'</td>
<td>Reserved (should be set off)</td>
</tr>
</tbody>
</table>

The above options are fully documented under their option name.

CBLVCSW3=X'xx'

A run time override for the CBLVCSW3 general options.

The default can be set using the CBLVCSW3 (CBL VCat SWitch 3) field in CBLNAME, which is supplied containing X'00'. Byte CBLVCSW3 is mapped as follows:

<table>
<thead>
<tr>
<th>X'80'</th>
<th>on = NOMOUNT option</th>
<th>off = MOUNT option</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'40'</td>
<td>on = long filenames overwrite following fields</td>
<td></td>
</tr>
<tr>
<td>X'20'</td>
<td>on = NOINDEX option</td>
<td>off = INDEX option</td>
</tr>
<tr>
<td>X'10'</td>
<td>on = AVRL option</td>
<td>off = LMAX option</td>
</tr>
<tr>
<td>X'08'</td>
<td>on = NOPSEV option</td>
<td>off = PRTSEV option</td>
</tr>
<tr>
<td>X'04'</td>
<td>on = NOPCAP option</td>
<td>off = PRTCAP option</td>
</tr>
<tr>
<td>X'02'</td>
<td>on = NOPJCL option</td>
<td>off = PRTJCL option</td>
</tr>
</tbody>
</table>
CBLVCSW4=X’xx’  (LISTVCAT,LISTVTOC OPTION)

LC  DD=CBLV22  CBLVCSW4=X’08’  * Use BUFND on TUNE JCL

A run time override for the CBLVCSW4 general options.

The default can be set using the CBLVCSW4 (CBL VCat SWitch 4) field in CBLNAME, which is supplied containing X’00’. Byte CBLVCSW4 is mapped as follows:

| X’80’ | on = CBLVCAT and CBLVTOC take no card input. |
| X’40’ | Reserved (should be set off). |
| X’20’ | VM users accessing MVS non-ICF catalogs require this bit on for the DEFINE option. |
| X’10’ | Reserved (should be set off). |
| X’08’ | on = Use BUFND on TUNE JCL Overrides. |
| X’04’ | on = Force cancel when VSE/SP 2 or higher. |
| X’03’ | on = Suppress console messages on error. |

The above options are fully documented under their option name.

CBLVCSW5=X’xx’  (LISTVTOC OPTION)

LV  VOL=CBLT01  CBLVCSW5=X’90’  * Sort by DATE

A run time override for the CBLVCSW5 SORT order.

The default can be set using the CBLVCSW5 (CBL VCat SWitch 5) field in CBLNAME, which is supplied containing X’00’. Byte CBLVCSW5 is mapped as follows:

| X’A0’ | sort by SIZE (descending) |
| X’90’ | sort by DATE (descending) |
| X’04’ | sort by EXPIRY |
| X’02’ | sort by DSN |
| X’01’ | sort by EXTPRIME |
| X’00’ | sort by EXTENT (default in CBLNAME ASSEMBLE) |

CBLVCSW6=X’xx’  (LISTVCAT OPTION)

LC  DD=CBLV22  CBLVCSW6=X’08’  * Create REORGANISATION Deck

A run time override for the CBLVCSW6 options.

The default can be set using the CBLVCSW6 (CBL VCat SWitch 6) field in CBLNAME, which is supplied containing X’00’. Byte CBLVCSW6 is mapped as follows:

| X’80’ | on = DEFINE option off = NODEFINE option |
| X’40’ | on = GDGRPT option off = NOGDGRPT option |
| X’20’ | on = GDGDSN option off = NOGDGDSN option |

The above options are fully documented under the relevant option name.

| X’10’ | on = Suppress Catalog name |
| X’08’ | on = Create Reorganisation deck |
| X’04’ | on = Use disk for Reorganisation work file |

The above switch settings are all documented under IDCAMS DEFINE and Reorganisation in Guide to LIST Output.

| X’02’ | on = Show the actual dataset name of the catalog above the DSN column for Standard and Customised reports. |
| X’01’ | on = Reserved (should be set off). |
CBLVCSW7=X'xx'

(LISTVCAT OPTION)

OPTION DEFINE  CBLVCSW7=X'04'  * No AIX decks

A run time override for the CBLVCSW7 DEFINE options.

The default can be set using the CBLVCSW7 (CBL VCat SWitch 7) field in CBLNAME, which is supplied containing X'00'. Byte CBLVCSW7 is mapped as follows:

<table>
<thead>
<tr>
<th>X'80'</th>
<th>on = PATH parameters suppressed</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'40'</td>
<td>on = BLDINDEX parameter(s) suppressed</td>
</tr>
<tr>
<td>X'20'</td>
<td>on = Suppress DEFINE comments</td>
</tr>
<tr>
<td>X'10'</td>
<td>on = Include Notes</td>
</tr>
<tr>
<td>X'08'</td>
<td>on = CLUSTER parameter(s) suppressed</td>
</tr>
<tr>
<td>X'04'</td>
<td>on = AIX parameter(s) suppressed</td>
</tr>
<tr>
<td>X'02'</td>
<td>on = DELETE deck(s) not commented out</td>
</tr>
<tr>
<td>X'01'</td>
<td>on = Job control suppressed</td>
</tr>
</tbody>
</table>

CBLVCSW8=X'xx'

(LISTVCAT OPTION)

OPTION DEFINE  CBLVCSW8=X'10'  * Suppress Catalog from the DELETE deck.

A run time override to suppress Catalog details from the DELETE parameters (This should be used with care).

The default can be set using the CBLVCSW8 (CBL VCat SWitch 8) field in CBLNAME, which is supplied containing X'00' (Catalog details included). Byte CBLVCSW8 is mapped as follows:

| X'10' | on = Catalog details suppressed in DELETE parameters |

All other bits are reserved and should be set to zero.

See also IDCAMS DEFINE and Reorganistaion in Guide to LIST Output.

CBLVCSW9=X'xx'

(LISTVCAT, LISTVTOC OPTION)

OPTION SHR  CBLVCSW9=X'05'  * SHR in standard report, 2-digit years & ISC.

A run time override for the CBLVCSW9 general options.

The default can be set using the CBLVCSW9 (CBL VCat SWitch 9) field in CBLNAME, which is supplied containing X'00'. Byte CBLVCSW9 is mapped as follows:

<table>
<thead>
<tr>
<th>X'10'</th>
<th>on = Suppress overtype of “X” in LISTVCAT TYPE field for SMS Extended and VSE VSAM EXTRALARGEDATASET files.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'08'</td>
<td>on = Display LISTVCAT devices in hexadecimal representation.</td>
</tr>
<tr>
<td>X'04'</td>
<td>on = Use VSAM In-Storage Catalog processing. (Option ISC)</td>
</tr>
<tr>
<td>X'02'</td>
<td>on = Display Local Timestamp based on Time Zone displacement.</td>
</tr>
</tbody>
</table>
| X'01' | on = Display 2 digit year in VTOC and catalog report date fields. (Otherwise 4 digit year).
Also includes S/C or SHR in the standard catalog report. |

The above options are fully documented under their option name.

All other bits are reserved and should be set to zero.
CI/CA (LISTVCAT OPTION)

**LC**
**OPTIONS** CI/CA UNUSED  

* Choose print columns

**SUBSET** UNALLOC=100  

* Select if 100 trks free

If **EXCPS** is the current installation default, CI/CA can be used to display the column containing the number of Control Intervals per Control Area (CI/CA) instead of the column containing number of Executed Channel Programs (**EXCPS**). CI/CA and if **EXCPS** are mutually exclusive options.

In the CBL supplied version of CBLNAME, **EXCPS** is the default (**CBLVCSW1** bit 'X'01' is on). CI/CA can be made the default by changing this setting.

See **REPORT VCAT CI/CA** below for an explanation of warning indications.

---

**CI/CA (nnn)**

**REPORT VCAT DSN CI/CA EXCPS**

Display the number of Control Intervals per Control Area.

The column width is 6, the format is ‘+nnnn*’ and the heading is **CI/CA** (**CI/CA** forms part of the **LISTVCAT** standard report, if option **CI/CA / EXCPS** is set to **CI/CA**).

**Note**

An asterisk appended to the right of the numeric value (nnnn*), indicates that the number of CIs in the CA, reserved to accommodate CI splits, is greater than the percentage requested when the file was defined. This generally happens because freespace in a CA has to be an integer number of CIs.)
**CIMAX=nnnn**

| LC  | DD=CBLV01 | TUNE CIMIN=2048 CIMAX=4096 |

Limits the maximum data CI size when tuning.

It should be used when the file being tuned is restricted to a specific or maximum CI size (either because it is used on-line, or restricted by another program).

It can also be used for modelling new files intended for on-line use.

**Notes**

For on-line files you may prefer to use the **CBLVCOn** fields (or run-time OPTIONS) which allow simultaneous tuning of batch and on line files.

**CIMAX** may be specified together with **CIMIN** to set a range of allowable CI sizes (Specifying **CISIZE=n1,n2** is equivalent and more convenient).

**CIMAX** and **CIMIN** are mutually exclusive of **CISIZE** (See **Tuning Considerations in Guide to VSAM Tuning** for more information).

---

**CIMIN=nnnn**

| LC  | DD=CBLV01 | TUNE CIMIN=2048 CIMAX=4096 |

Limits the minimum data CI size when tuning.

It should be used when the file is being tuned exclusively for the batch environment, but can also be used for modelling new files intended for batch use.

**Notes**

**CIMIN** may be specified together with **CIMAX** to set a range of allowable CI sizes (Specifying **CISIZE=n1,n2** is equivalent and more convenient).

**CIMAX** and **CIMIN** are mutually exclusive of **CISIZE** (See **Tuning Considerations in Guide to VSAM Tuning** for more information).

---

**CISIZE (nn) - REPORT VCAT**

| REPORT VCAT | DSN TYPE NRECS SORTD CISIZE LMAX AVRL |

Display the control interval size.

The column width is 7, the format is ‘+nnnnn*’ and the heading is **CISIZE** (**CISIZE** forms part of the standard report).

**Notes**

An asterisk appended to the right of the numerical value (nnnnn*) indicates that the CISIZE is insufficient to hold at least two records and the defined CI Freespace.

A plus sign either side of the value (+nnnnn+), indicates that the index control interval size is greater than required. (The CBLVCAT calculation is based on the assumption that normal key compression will take place - See **KEY COMPRESSION** in Additional VSAM Information).

The CI size influences (among other things) the proper **BUFSP** recommendation for a tuning run (See **CI SIZE** in Additional VSAM Information and **TUNE block** in Guide to VSAM Tuning for more information on CI size).

---

**CISIZE (nn) - REPORT VTOC**

| REPORT VTOC | DSN TYPE SORTD CISIZE START ALLOC |

Display the control interval size (FBA disks only).

The column width is 7, the format is ‘+nnnnn*’ and the heading is **CISIZE** (**CISIZE** forms part of the the standard report for FBA disks).
Notes
The CI size also affects the minimum file size, as the file must comprise whole CIs.
The minimum file size is 2 CIs, 1 for data and 1 for Software End Of File (SEOF).

CISIZE=KEEP

Keep the DEFINED data CI size for this tuning run.
This should be used when other program products restrict the allowable CI size of the file(s) they process.

Notes
This parameter works best when a single file is chosen via the subset KEY=string sub-parameter.
CISIZE=KEEP is mutually exclusive of CIMIN, CIMAX and CISIZE=n1,n2 (See Tuning Considerations in Guide to VSAM Tuning for more information).
It can also be used to stop CBLVCAT recommending that SPANNED files become NON SPANNED.

CISIZE=n1,n2

Limits the data CI size tuning recommendations to the range of sizes within 'n1' and 'n2' inclusive.

Notes
The 'n2' value may be omitted, in which case only the 'n1' value is used for tuning (This is equivalent to specifying the same value for 'n1' and 'n2').
CISIZE=n1,n2 is mutually exclusive of CIMIN, CIMAX and CISIZE=KEEP (See Tuning Considerations in Guide to VSAM Tuning for more information).

CLASS=n

For VSE only, used to select files that reference a particular SPACE CLASS.
Use option S/C for displaying the class of the primary extent.

COMPONENT (nn)

For LISTVCAT VVDS only, display the VSAM component name.
The column width is 44 and the heading is COMPONENT.
Example: LISTVCAT COMPONENT field (CBL Ref: vbxv5f1)

Notes

All VSAM clusters have a DATA component but KSDS and AIX clusters also have an index component. The component name can be defined explicitly by the user, or be generated by IDCAMS based on the cluster name.

In MVS these component names are also held in the VTOC.

The example above illustrates the difference between the dataset name (heading VVDS volnnn), component name (heading COMPONENT) and the entry name (heading ENTRY).

COMPRESSED (LISTVCAT SUBSET)

CMP

LC KEY=CBL.CBLI.MBRLIST COMPRESSED

For MVS, selects all data sets that have been defined with SMS DATACLASS attribute COMPACTION.

For VSE, selects all VSAM KSDS clusters that have been defined with IDCAMS DEFINE option COMPRESSED.

CREATED (nn) (REPORT VTOC)

Display the date when the Format 1 label was created.

The column width is 10, the format is ‘ccyy/mm/dd’ and the heading is CREATED (CREATED also forms part of the standard report).

Notes

If CBLVCSW9=X’01’ is set on (CBLNAME option V2digitYear=Yes) then the format of the CREATED field is ‘yy/mm/dd’.

CYL/HD (nn) (REPORT VTOC)

Display the low and high extent limits (CKD devices only).

The column width is 16, the format is ‘+cccc/hh+cccc/hh’ and the column heading is CYL/HD LIMITS (CYL/HD also forms part of the standard report for CKD devices only).
**CYLMAX=nnn**

Limits the maximum size of the PRIME allocation recommended by CBLVCAT.

If this sub-parameter is not supplied, the default is the device capacity (See Guide to VSAM Tuning for more information).

**DDNAME=fname**

Selects the required catalog by referencing the MVS DD Statement, the CMS DLBL, or the VSE DLBL.

**Notes**

If DDNAME is omitted, LISTVCAT will operate on the catalog processed by the previous LISTVCAT operation (This includes the last catalog processed by a REF=ALL instruction.) If no previous operation, the catalog name held in the field CBLVCFN in CBLNAME is used (default is USYSUC).

For ICF catalogs the report is based on the BCS, however, reporting on the VVDS is also possible (see VVDS=volser).

For MVS systems only it is possible to use the REF parameter to select the catalog.

New users should refer to the Introduction and Guide to List Output for examples and more information.

**DDNAME=fname**

For MVS systems only, used to select the required disk volume by referencing a DD statement.

**Notes**

Selecting a disk volume is mandatory on the first LISTVTOC command, but optional on any LISTVTOC commands that follow. If it is omitted the VTOC report produced will be of the same volume as the previous LISTVTOC command. The omission of this parameter is useful when different report variations are required (such as SORT sequence) on the same VTOC.

See also the DEV and VOL parameters.

New users should refer to the Introduction and Guide to List Output for examples and more information.

**DEFINE**

If NODEFINE is the current installation default, DEFINE can be used to produce an output file of IDCAMS DELETE/DEFINE statements for all selected datasets (AIX/ESDS/KSDS/RRDS only). (DEFINE and NODEFINE are mutually exclusive options).

In the CBL supplied version of CBLNAME, NODEFINE is the default (CBLVCSW6 X'80' bit off). DEFINE may be made the default by changing this switch.

**Notes**

The file will also contain any associated PATH and BLDINDEX values.

The level of detail included in the IDCAMS DEFINE parameters is controlled by the switches CBLVCSW6, CBLVCSW7 and CBLVCSW8 (see IDCAMS DEFINE and Reorganisation in Guide to VSAM Tuning).
**DEFINED (nn)** (REPORT VCAT)

Display the date the file was defined.

The column width is 11 and the heading is **DEFINED** (**DEFINED** also forms part of the standard report if the option **TIMESTAMP/DEFINED** is set to **DEFINED**).

Where included as part of the LISTVCAT standard report, **DEFINED** has the format ‘*ccyy/mm/dd*. However, if specified on REPORT as part of a customised report, then the format is ‘ccyy/mm/dd’ with ‘*’ (asterisk) following the date as opposed to preceding it. This is so that **SORT** on the **DEFINED** field is reliable.

**Notes**

If **CBLVCSW9=X’01’** is set on (**CBLNAME** option V2digitYear=Yes) then the format of the **DEFINED** field is ‘*yy/mm/dd’ or ‘yy/mm/dd*’ for standard or customised reports respectively. Note that this option will also include report field **SHR** or **S/C** as part of the standard report.

*ccyy/mm/dd
An asterisk preceding the defined date (or following it in a customised report) indicates today’s date.

***nnn CI SPLITS** and ***nnn CA SPLITS**
These warnings are displayed in the **DEFINED** column for a standard report. They indicate that the the appropriate CI and CA splits warning threshold has been reached (**CBLVCSCI** or (**CBLVCSCA**).

**DEFINED** (LISTVCAT OPTION)

If **TIMESTAMP** is the current installation default, **DEFINED** can be used to display the file definition date, instead of the date the file was last closed by an operation that may have altered its contents (**TIMESTAMP**). **TIMESTAMP** and **DEFINED** are mutually exclusive OPTIONS.

In the CBL supplied version of **CBLNAME**, **TIMESTAMP** is the default (**CBLVCSW1** bit X’02’ is on). **DEFINED** may be made default by changing this switch.

**Notes**

The setting of this option also governs which dates (**TIMESTAMP** or **DEFINED**) are to be used for the selection of a **HIDATE**, **LODATE** SUBSET.

**DEL** (COMMAND)

**DEL**

DEL DEV=161
VOL=PACK11
DSN='TEMP CBL'
FAIL=IGN

* Choose device
* Check volume (optional)
* Quotes required (for blanks)
* Ignore errors

For **VSE** systems only, the **DEL** command allows deletion of a file (**DSN=xxx**) or all the files on a volume (**ALLFILES=YES**).

**Note**

Select and optionally check a device using one, or any combination of the **DEV**, **SYS** and **VOL** parameters. The **OWN** parameter may also be used as a means of checking the device.

**DEV=ALL** (LISTVTOC)

**VOL=**

ALL

LV
EJ=YES

* A new page for each report
LV
DEV=ALL

* All assigned drives
VSE systems

A VTOC report is produced for every active disk drive in the system (i.e. Operational (or DVCUP), READY (online) and not assigned IGNORE). To obtain VTOC listings of all assigned disk drives use SYS=ALL.

MVS systems

A VTOC report is produced for each DASD device on the system belonging to the esoteric group SYSALLDA.

The storage and run-time considerations can make this less attractive to the larger user. These are due to the fact that CBLVCAT stores all VTOC information in preparation for further LISTVTOC commands referencing the same volumes (subsequent LISTVTOC operations, within the same invocation will benefit from significant run-time improvements). Users with large systems should consider DEV=unitname.

With MERGE

Use of DEV=ALL with MERGE in a REPORT VTOC SORT allows the user to obtain a single combined report of all VTOCs sorted in any order (See example 15 in Guide to LIST Output).

Note

If the synonym ALL is used to produce the report, it must be the first parameter following the LISTVTOC command.

---

**DEV=unitname**

<table>
<thead>
<tr>
<th>(LISTVTOC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LV EJ=YES</td>
</tr>
<tr>
<td>DEV=3380DASD</td>
</tr>
</tbody>
</table>

For MVS systems only, this will result in a VTOC report for a user defined esoteric group of units (generic subset).

---

**DEV=cuu**

<table>
<thead>
<tr>
<th>(DEL,LISTVTOC,MOD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISTVTOC DEV=130</td>
</tr>
<tr>
<td>VOL=SYSWK1</td>
</tr>
<tr>
<td>MOD DEV=135</td>
</tr>
<tr>
<td>DSN=ABC.PROD</td>
</tr>
<tr>
<td>NEWDSN=ABC.TEST</td>
</tr>
<tr>
<td>EXP=TEMP</td>
</tr>
<tr>
<td>DEL DEV=135</td>
</tr>
<tr>
<td>DSN=ABC.PROD</td>
</tr>
</tbody>
</table>

For VSE systems only, the DEV and/or SYS and/or VOL parameters determine the current volume. When a combination of two or more of these parameters is specified, the order of checks for conformance are as follows:

**DEV and SYS**
The assignment specified by SYS must match the cuu specified by DEV otherwise ERROR 28 is incurred.

**DEV and VOL**
The volser specified by VOL must match the volume mounted on the cuu specified by DEV otherwise the OPERATOR is prompted to mount the correct volume.

**SYS and VOL**
The volser specified by VOL must match the volume mounted on the DASD assigned to by SYS otherwise the OPERATOR is prompted to mount the correct volume.

If a check fails then the operation will fail, with subsequent processing depending on the FAIL parameter.

Notes

If DEV, SYS and VOL are omitted, the drive will be that last used or will default to SYS=0 if no previous device specified.

Within a single invocation of CBLVCAT, a second or subsequent LISTVTOC operation on the same volume(s) will benefit from significant run-time improvements, as it uses previously stored information.

See also the OWN parameter for checking that the correct volume is mounted.
DEV=nnnn

Directs CBLVCAT to base its tuning recommendations on the device type specified.

It is designed to be used for modelling new files and for migrating existing files to new disks. Disk devices supported are:

0671 3330 3340 3350 3380 3390 3310 3370 3375 9332 9335 9336 9345

DSN=xxx.xxx.xx.etc

For VSE systems only, selects the file to be deleted or modified.

Notes

If the DataSet Name contains blanks, it must be supplied within single quotation marks.

Select and optionally check a device using one, or any combination of the DEV, SYS and VOL parameters. The OWN parameter may also be used as a means of checking the device.

DSN (nn)

Display the VSAM filename.

The column width is 44. For ICF Catalogs the heading is 'ICF CAT vvvvv (nnnn)' whereas for non-ICF Catalogs the heading is 'USERCAT vvvvv (nnnn)' (where 'vvvv' is the volume serial number of the catalog disk and 'nnnn' is its device type).

Example: LISTVCAT associations (CBL Ref: vmxassc)
When the **MERGE** feature is used the heading becomes **DATASET NAME**.

The standard heading (e.g. 'USERCAT CBLV03 (3380)') can be replaced by the actual data set name of the catalog being processed, by setting **CBLVCSW6 X'02'** on.

If **DSN** is supplied with a **numeric value** less than 44 and no **REPORT SORT** parameter, any names that exceed the column width will be printed in full with data for all subsequent columns printed on the next line of the report (The **SORT** parameter causes the DSN to be **truncated**).

**ASSOC** indicates that the file has ASSOCIATIONS (other than data or index). Printing association details can be enabled/disabled using the **ASSOC/ NOASSOC** option, or by changing **CBLVCSW1** in your **CBLNAME** program.

**DSN** also forms part of the standard report with a column width of 22. For this report, the printing of names longer than these values is governed by the **OVLAY/ NOOVLAY** option.

---

**EJECT**

(ECOMMAND,DEL,LISTVCAT,LISTVTOC,MOD)

**EJECT=**YES **EJ**

**LC DD=CBLV01** **EJECT=YES**

**LC DD=CBLV02** **EJECT**

**LV DEV=230**

**EJECT** allows the user to control the page skipping.

**EJECT** is a command if it is the first control word on a statement and a new page is forced immediately, otherwise it is a parameter and a new page is forced immediately after the printing of the operation on which **EJECT** was specified (**EJECT** is not printed).

---

**ENTRY** (nn)

(REPORT VCAT)

**ENTRY** (nn)

**REPORT VCAT DSN TYPE COMPONENT ENTRY**

For **LISTVCAT VVDS=volser** only, used to display the VSAM entry name of the component (For AIXs the associated KSDS dataset name is displayed).

The column width is 44 and the heading is **ENTRY**.

**Notes**

The **ENTRY** name is generated by IDCAMS when the dataset is defined and, for KSDS, ESDS, RRDS and LDS components, it is the same as the cluster name.

See **COMPONENT** for a comparison between **DSN**, **COMPONENT** and **ENTRY** names.

---

**EXCPS**

(LISTVCAT OPTION)

**LC DD=CBLV01** **OPTION EXCPS**

If **CI/CA** is the current installation default, **EXCPS** can be used to display the column containing the number of EXecuted Channel Programs (**EXCPS**) instead of the column containing the number of CIs per CA (**CI/CA**). **EXCPS** and **CI/CA** are mutually exclusive options.

In the CBL supplied version of **CBLNAME**, **EXCPS** is the default (**CBLVCSW1** bit **X'01'** is on). **CI/CA** can be made the default by changing this setting.

See Notes in **EXCPS REPORT VCAT** below.
EXP=PERM (MOD)
MOD SYS=04 VOL=SYSWK2 * Choose device on SYS004 and verify as VOL=volser.
DSN=PROD.ACC.XYZ EXP=PERM * Change to 1999/366.

For VSE systems only, change a file expiry date to 1999/366, thus making it permanent.

Notes
The file data set name must be supplied via the DSN parameter.
Select and optionally check a device using one, or any combination of the DEV, SYS and VOL parameters. The OWN parameter may also be used as a means of checking the device.

EXP=TEMP (MOD)
MOD SYS=04 VOL=SYSWK2 * Choose device on SYS004 and verify as VOL=volser.
DSN=TEST.ACC.XYZ EXP=TEMP * Change to 1901/001.

For VSE systems only, change a file expiry date to 1901/001, thus making it temporary.

Notes
The file data set name must be supplied via the DSN parameter.
Select and optionally check a device using one, or any combination of the DEV, SYS and VOL parameters. The OWN parameter may also be used as a means of checking the device.

EXPD (LISTVTOC OPTION)
LV VOL=CBLT01 OPTION EXPD * Override NOEXPD in CBLNAME

If NOEXPD is the current installation default, EXPD can be used to display ‘EXPD’ (instead of the actual expiry date) for files that have expired or have been defined without an expiry date.
EXPD and NOEXPD are mutually exclusive options.
In the CBL supplied version of CBLNAME, EXPD is the default (CBLVCSW2 bit X'04' is off). NOEXPD can be made the default by changing this setting.

Example: LISTVTOC with EXPD OPTION (CBL Ref: vmxexp1)

Example: LISTVTOC with EXPD OPTION (CBL Ref: vmxexp1)
Example: LISTVTOC with NOEXPD OPTION (CBL Ref: vmxexp2)

Notes

If EXPD is in force and the FREETAB option is used, the space occupied by expired files is included in the Freespace table with EXPD appended to it.

If NOEXPD is in force, the space occupied by expired files is not included in this table.

---

### EXPIRES (nn) - REPORT VTOC

Display the expiry date of the file. i.e. the date when extents in the VTOC belonging to a file may be overwritten without a warning message.

The column width is 10 and the column heading is **EXPIRES** (EXPIRES also forms part of the standard report).

Where included as part of the LISTVTOC standard report, EXPIRES has the format ‘ccyy/mm/dd’. However, if specified on REPORT as part of a customised report, then the format is ‘ccyy/mm/dd*’ with ‘*’ (asterisk) following the date as opposed to preceding it. This is so that SORT on the EXPIRES field is reliable.

Notes

If CBLVCSW9=X’01’ is set on (CBLNAME option V2digitYear=Yes) then the format of the EXPIRES field is ‘yy/mm/dd’ or ‘yy/mm/dd*’ for standard or customised reports respectively.

*ccyy/mm/dd
An asterisk preceding the expiry date (or following it in a customised report) indicates that an expiry date of 1999/12/31 which has been calculated from a retention period value is not considered permanent by the operating system. Although this may be intentional, it is highlighted so that it may be checked by the user to avoid files being silently deleted on or after 31st December 1999.

The contents of this column are affected by the **EXPD/ NOEXPD** and **PERM/ NOPERM** and options.

Possible entries in this column are:

1. **The actual expiry date.**
   Displayed for expired and/or permanent files, if the NOEXPD and/or NOPERM options are active.

2. **EXPD**
   No expiry date has been defined for the file, the file is temporary (expiry date of 01/001 = 1901/01/01) or the expiry date is on or before the report heading date. (The EXPD/NOEXPD option has to be set to EXPD).

3. **TODAY**
   The file expired on the day when the report was produced (The EXPD/NOEXPD option has to be set to EXPD).

4. **PERMANENT**
   The file is permanent i.e. created with an explicit expiry date of 1999/365 or 1999/366. An expiry date of 1999/12/31 which has been calculated from a retention period value is not considered permanent by the operating system. (The PERM/NOPERM option has to be set to PERM).
### EXPIRES (nn) - REPORT VCAT

Display the expiry date of the file as defined by the IDCAMS DEFINE TO(...) parameter.

<table>
<thead>
<tr>
<th>CBLVCAT REL 2.12 AT CBL - Bridgend UK (Internal Only)</th>
<th>2009/08/20 16:30 PAGE 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>option pw=133 ** CBL.VVC.CTL(EXPIRES) ***</td>
<td>L=001 +++ 2001/05/04 16:38:02 (NBJ)</td>
</tr>
<tr>
<td>report vcat dsn type EXPIRES</td>
<td></td>
</tr>
<tr>
<td>listvcat ref=usercat.cblcat type=m novol noassoc noindex</td>
<td>User catalog, VSAM + No volume/assoc/index info.</td>
</tr>
<tr>
<td>ign=/CTRDB key=CBL</td>
<td>All DSN &quot;CBL.&quot; without &quot;.CTRDB..&quot;</td>
</tr>
<tr>
<td>** CBL.VVC.TMP **</td>
<td>2022/01/11</td>
</tr>
</tbody>
</table>

Example: Catalog Report with REPORT field EXPIRES (CBL Ref: vvexpire)

The column width is 10, the format is 'ccyy/mm/dd' and the column heading is **EXPIRES**.

### Notes

Valid expiry dates such as yyyy/366, or yyyy/999 which cannot be converted from Julian to International format, will be printed unchanged.

### EXT (nn) - REPORT VCAT

Display characteristics of MVS data sets with SMS extended attributes and VSE VSAM files defined with IDCAMS DEFINE parameter EXTRALARGEDATASET.

The column width is 11, the format is 'xxx(x)x(nn)' and the column heading is **EXT-ATTRIB**.

Possible entries in this column are as follows:

<table>
<thead>
<tr>
<th>Position</th>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X</td>
<td>Data set is defined as being Extended.</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
<td>Data set is defined as being of Extended Addressability or, for VSE, defined with EXTRALARGEDATASET. (Extended flag is also displayed).</td>
</tr>
<tr>
<td>3</td>
<td>C(x)</td>
<td>Data set is defined as being Compressed. For MVS, the value in parentheses indicates the type of compression. Possible values are: C(G) Generic DBB (Dictionary Building Block) compression.</td>
</tr>
</tbody>
</table>

### Example

```
Example: Catalog Report with REPORT field EXPIRES (CBL Ref: vvexpire)
```

The column width is 10, the format is 'ccyy/mm/dd' and the column heading is **EXPIRES**.

### Notes

Valid expiry dates such as yyyy/366, or yyyy/999 which cannot be converted from Julian to International format, will be printed unchanged.
• **C(T)** Tailored compression.
• **C(R)** Compression Rejected.

For VSE, the value in parentheses indicates the compression status. Possible values are:

• **C(P)** Compression Pending.
• **C(A)** Compression Active.
• **C(R)** Compression Rejected.
• **C(U)** Compression Undetermined.

| 7 | S(nn) | Data set is defined as being Striped and has nn stripes (where nn>01). |

---

### EXT-ADDR (LISTVCAT SUBSET)

**EXTA**

**EXTENDED** (VSE Only)

**EXT** (VSE Only)

**XVSAM** (VSE Only)

| LC | KEY=CBL.CBLI.MBRLIST | EXT-ADDR | TYPE=C |

For MVS, selects all data sets that have been defined with SMS DATACLASS attribute Extended Addressability.

For VSE, selects all VSAM KSDS clusters that have been defined with IDCAMS DEFINE option EXTRALARGEDATASET.

### EXTENDED (LISTVCAT SUBSET)

**EXT**

| LC | DD=VSESPUC | EXTENDED |

For MVS, selects all data sets (VSAM and non-VSAM) that have been defined with SMS DATACLASS attribute DSNTYPE=EXT.

For VSE, EXTENDED is a synonym for **EXT-ADDR**.

### EXTNO=nn (LISTVTOC SUBSET)

**EXTNUM=nn**

| LV | VOL=CBLT01 | SUBSET | EXTNO=6 |

Selects files with at least nn extents.

### FAIL=xxx (LISTVCAT,LISTVTOC,DEL,MOD)

| DEL | DEV=161 | ALLFILES=YES | VOL=SYSWK1 | FAIL=IGNORE |
| MOD | DEV=104 | DSN=ABC.TEST | NEWDSN=DEF.TMP | FAIL=EOJ |
| LV | VOL=CBLT03 | LOEXP=93/10/27 | FAIL=CANCEL |

Used to select the action following an operation failure.

**CANCEL**

End processing immediately with an appropriate message and a non-zero return code (default).

**IGNORE**

Ignore the error and continue with the next operation.

**EOJ**

End processing immediately with an appropriate message and return code.
FREE (LISTVTOC OPTION)

If FREETAB (or NOFREE) is the current installation default, FREE can be used to display Freespace extents as they occur, within the main body of the report (FREE, FREETAB and NOFREE are mutually exclusive options).

In the CBL supplied version of CBLNAME, NOFREE is the default (CBLVCSW2 bit X'10' is off). FREE can be made the default by changing this setting (CBLVCSW2 bit X'10' on and bit X'08' off).

<table>
<thead>
<tr>
<th>LISTVTOC VOL=CBLT03 FREE</th>
</tr>
</thead>
<tbody>
<tr>
<td>o</td>
</tr>
<tr>
<td>VTOC OF CBLT03 CISIZE START ALLOC USED TYPE EXPIRES BL</td>
</tr>
<tr>
<td><em>FREE</em> 557256 286 13 FBA</td>
</tr>
<tr>
<td>TEST.SCRN.LIBRARY.4 512 154788 1500 PERMANENT</td>
</tr>
<tr>
<td>TEST.SOURCE.LIBRARY.4 512 156288 36000</td>
</tr>
<tr>
<td>Z9999994.VSAMDSPC.T9A1BDD6.T8976140 192288 139536</td>
</tr>
<tr>
<td>Z9999992.VSAMDSPC.T9A1BDD9.T8014540 331824 744 VSAM PERMANENT</td>
</tr>
<tr>
<td>MASTER SORT TABLE 4608 557542 104098 PERMANENT</td>
</tr>
</tbody>
</table>

Example: LISTVTOC FREE option (CBL Ref: vmxfree)

Note

SORT=EXP/DATE/DSN used in conjunction with FREE will not report on freespace, as freespace does not have a valid DSN or any create/expiry date information.

FREEBYTES (nn) (REPORT VCAT)

<table>
<thead>
<tr>
<th>REPORT VCAT DSN 3D HIUSERBA HIALLRBA FREEBYTES</th>
</tr>
</thead>
</table>

Display the Freespace Bytes value (as required for investigative purposes).

The column width is 11, the format '+nnnnnnnnnn' and the heading is FSPC BYTES.

Notes:

The FREEBYTES value is calculated based on FREESPACE, HIUSERBA and HIALLRBA values.

The FREEBYTES value is used by CBLVCAT in calculating its tuning recommendations for Catalog reports with TUNE.

FREETAB (LISTVTOC OPTION)

<table>
<thead>
<tr>
<th>LV VOL=CBLT03 FREETAB EXPD * Freespace, includes expired files</th>
</tr>
</thead>
<tbody>
<tr>
<td>LV VOL=CBLT03 FREETAB NOEXPD * Freespace, excludes expired files</td>
</tr>
</tbody>
</table>

If FREE (or NOFREE) is the current installation default, FREETAB can be used to display Freespace information separately at the end of the report (FREETAB, FREE and NOFREE are mutually exclusive options).

In the CBL supplied version of CBLNAME, NOFREE is the default (CBLVCSW2 bit X'10' is off). FREETAB can be made the default by changing this setting (CBLVCSW2 bit X'10' on and bit X'08' on).
The Freespace table will be printed in ascending extent order unless \texttt{SORT=SIZE} is in effect, in which case the display will be in descending order.

As well as the total number of tracks/blocks used, the percentage this constitutes is also displayed. When this percentage is flagged with asterisks, the user defined tracks full percentage threshold (see \texttt{CBLVCPC} in \texttt{CBLVCPC}) has been reached. This is accompanied by a \texttt{WARN 003} message (\texttt{Return Code 06} also produced).

The Freespace table is also affected by the \texttt{EXPD/ NOEXPD} option. If \texttt{EXPD} is in force, expired files are displayed as a part of the table with \texttt{EXPD} appended to the right of the value whereas if \texttt{NOEXPD} is in force they are not included.

See also example 18 - ‘Free space in all VTOCs’ in Guide to List Output.

### FRSP (nn) (REPORT VCAT)

Display the KSDS/AIX distributed Freespace percentage fields (bytes/CI and CIs/CA).

The column width is 7, the format is ‘-\texttt{nn+nn}’ and the heading is \texttt{FRSP CI CA} (\texttt{FRSP} also forms part of the standard report).

Freespace is reserved during an initial file load and also during sequential insertion if CI or CA splits occur (see CI splits, CA splits and Freespace in Additional VSAM Information).
<table>
<thead>
<tr>
<th>USERCAT</th>
<th>CBLV11 (3380)</th>
<th>TYPE</th>
<th>NRECS</th>
<th>PCNT</th>
<th>----</th>
<th>ALLOC TRACKS ----</th>
<th>FRSP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>o</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>o</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>o</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>o</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>o</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notes**

The value displayed is the amount specified at define time, not necessarily the actual amount reserved.

Both entries left blank indicates that both FRSP CI and FRSP CA are zero. This means that no space has been reserved for adding records to a KSDS file and inserts will cause immediate CI and CA splits (very costly). When the file in question is static (i.e. no insertions take place) it is correct to specify no Freespace.

An asterisk to the right of the FRSP CI value (nn*) indicates the space reserved is unusable as it is insufficient to support the insertion of at least one record.

An asterisk to the right of the FRSP CA value (nn*) indicates that either VSAM has reserved twice the requested Freespace, or the amount reserved is 20% higher than that DEFINED.

---

**FRSPCA=nn**

(LISTVCAT TUNE)

```
LC DD=TESTCAT   KEY=CURRENT.FILE.2
TUNE   FRSPCI=20   FRSPCA=5
```

Sets the percentage of CIs per CA to be left free during initial load (KSDS and AIX files).

This is supplied as an override to the current Catalog statistics and the tuning recommendations will reflect this override.

**Notes**

Ordinarily the GROWTH parameter should be used (see Growth and Freespace in Tuning Considerations).

FRSPCA may be used with FRSPCI, but neither may be used with GROWTH.

FRSPCA=0 and FRSPCI=0 (or GROWTH=0) should be specified for static KSDS files.

---

**FRSPCI=nn**

(LISTVCAT TUNE)

```
LC DD=TESTCAT   KEY=CURRENT.FILE.2
TUNE   FRSPCI=20   FRSPCA=5
```

Sets the percentage of CI space to be left free during initial load (KSDS and AIX files).

This is supplied as an override to the current Catalog statistics and the tuning recommendations will reflect this override.

See FRSPCA for notes on its use.

---

**GDGDSN**

(LISTVCAT OPTION)

```
LC DD=CBLI01
TYPE=G   GDGDSN
```

For use in conjunction with REPORT VCAT. If NOGDGDSN is the current installation default, GDGDSN can be used to append the GDG Dataset Name with the absolute generation number.
In the CBL supplied version of CBLNAME, NOGDGDSN is the default (CBLVCSW6 bit X'20' set off). GDGDSN can be made the default by changing this setting.

---

Example: REPORT VCAT GDGDSN (CBL Ref: vmxgdsn)

Notes

In this example the overflow information relating to each Catalog entry (i.e. the items which will not fit on a single line) is printed without the DSN (the DSN is only included once). See GDGRPT for information on printing the DSN on each line.

The GDGDSN is made up of the generation number (GGEN - in this example '0529' and '0530') and the version number (GVER - in this example '17' and '18'). The version number can run from 00 to (maximum GDG level)-1. (maximum GDG level is supplied at file definition time).

The specification of VCAT report parameter VOL1 with a numerical argument, causes as many VOL1 data items (serial numbers) to appear on a line as possible within the given field width (if no argument is supplied, the default is one data item per line).

GGEN, GVER and GMAX are all available as separate report fields.

---

GDGRPT (GDG repeat)

LC DD=CBLI01 TYPE=G GDGRPT

For use in conjunction with REPORT VCAT. If NOGDGRPT is the current installation default, GDGRPT can be used to repeat the unchanged GDG information on every line that contains overflow information for that file.

In the CBL supplied version of CBLNAME, NOGDGRPT is the default (CBLVCSW6 bit X'40' set off). GDGRPT can be made the default by changing this setting.

Note

This can be particularly useful for the post-processing of reports.

---

GGEN (nn) (GDG gen nos)

REPORT VCAT DSN 22 TYPE GMAX GGEN GVER VOL3 25

Display the generation numbers of GDG files (see GDGDSN Notes).

The column width is 4, the format is 'nnnn' and the column heading is GEN.

---

GMAX (nn) (GDG max level)

REPORT VCAT DSN 22 TYPE GMAX GGEN GVER VOL3 25

Display the maximum level of GDG files (see GDGDSN Notes).

The column width is 3, the format is 'nnn' and the column heading is MAX.
GROWTH=nn

Sets the percentage growth expected before the file(s) are next re-organised.

The value can be from 0 to 99 percent.

This is supplied as an override to the current Catalog statistics and the tuning recommendations will reflect this override.

Notes

If GROWTH is used, FRSPCA and/or FRSPCI cannot be used.

GROWTH=0 (or FRSPCA=0 and FRSPCI=0) should be specified for static KSDS files which experience no inserts.

For KSDS and AIX files, GROWTH affects the distributed free space only, thus assuming that the initial load will be for the number of records that currently exist (Use the RECORDS=nnn parameter in order to increase/decrease the initial file size).

For ESDS and RRDS files, GROWTH increases the size of the primary allocation, in order to prevent secondary allocations when records are added to the file.

See Growth and Freespace in Tuning Considerations.

GVER (nn)  (GDG vers no)

Display the version number of GDG files (see GDGDSN Notes).

The column width is 2, the format is 'nn' and the column heading is VER.

HEAD='string'

CBLVCAT's single line page heading may be replaced by a heading of your own choice (or suppressed altogether) using this parameter.

Notes

A null string as the argument to HEAD is not permitted.

If HEAD=NO is coded then the standard heading is suppressed (the time, date and page number (right adjusted) will still remain).

If the HEAD parameter is coded for the first report, then the CBLVCAT heading is overwritten before printing any control cards.

Multiple HEAD parameters are allowed. The argument of the last HEAD parameter encountered will be used for all headings from that point onwards.

Second and subsequent HEAD parameters will automatically force a new page for the new report, therefore, use of an EJECT parameter to force a new page is unnecessary.

The heading supplied (maximum length of 59 bytes) is left adjusted and underlined. The partition/operating system identifier and Job name/User Id are removed. The time, date and page number are right adjusted according to the heading width, however, they will be shortened or removed altogether for large HEAD strings with small heading widths.

HIALL RBA (nn)

Display the High Allocated Relative Byte Address (for investigative purposes).

The column width is 11, the format '+nnnnnnnnnn' and the heading is HI ALL RBA.
**Notes:**

For MVS data sets defined with SMS DATACLASS attribute "Extended Addressability" or VSE VSAM files defined with IDCAMS attribute EXTRALARGEDATASET, the HIALLRBA value is stored as a relative CI address instead of a relative byte address.

CBLVCAT converts a relative CI address to a relative byte address by multiplying the value by the CISIZE and displays the product in the HIALLRBA column.

The HIALLRBA value is used by CBLVCAT in calculating its tuning recommendations for Catalog reports with TUNE.

---

**HIBLK=nnnn**

| LISTVCAT | DD=CBLV03 | CBLVCEXT=5 | LOBLK=100000 | HIBLK=109999 |

For FBA Disks only (see HICYL or HITRK for CKD). HIBLK is used to select files whose extent(s) reside fully or partially below this block limit.

---

**Example:** LISTVCAT LOBLK/HIBLK subset (CBL Ref: vmxhibv)

---

**Notes**

HIBLK may be used with LOBLK to define upper and lower boundaries.

For **LISTVCAT** reports, the extent information is not normally displayed (Specifying CBLVCEXT=nn will display up to 'nn' extents per file **BUT ONLY** if they satisfy the LOBLK/HIBLK criteria).

For **LISTVTOC** reports, all extents are printed for each file that satisfy the HIBLK and/or LOBLK subset.

These rules also apply to **HICYL**, **LOCYL** and **HITRK, LOTRK**.
HICISZ=nnnn  (LISTVCAT SUBSET)

LC  DD=CBLI11   LOCISZ=2048   HICISZ=4096

Select files with a CI size less than or equal to the value specified.
HICISZ may be used with LOCISZ to restrict to a specified range of CI sizes.

HICYL=nnnn  (LISTVCAT, LISTVTOC SUBSET)

LV  DD=DD1   LOTRK=30   HICYL=10

For CKD Disks only (see HIBLK for FBA). HICYL is used to select files whose extent(s) reside fully or partially below this cylinder limit.

See Notes in HIBLK for further information.

HIDATE=date  (LISTVCAT, LISTVTOC SUBSET)

LC  DD=CBLI11   LODATE=2005/02/22   HIDATE=20090507
LV  DEV=ALL     LODATE=20070918   HIDATE=2008/09/17
LC  KEY=CBL.X232.REP   LODATE=-30   HIDATE=-1

Select files with a date less than or equal to the value specified.

Notes

Specifying LODATE and HIDATE together sets a range of dates.

Specifying the same date for LODATE and HIDATE limits the report to a single date.

The date may be: '+'/-ddd', 'ccymmd', 'ymmd', 'ccyy/mm/dd', 'yy/mm/dd' or a shorter form (Examples are: '+7', '-30', '20020703', '930506', '93/05/06', or '9305', '93/05', or '93').

The century for the 'ymmd' and 'yy/mm/dd' formats of the argument is interpreted using a sliding window in the range -79 to +20 relative to Current Date.

See Year 2000 Compliance for examples of 'Sliding Windows'.

The formats 'dd/mm/yy' and 'mm/dd/yy' are not supported and will result in an an ERROR 16.

The format '+'/-ddd' allows the date to be expressed as the number of days relative to the current date.

For LISTVCAT, if TIMESTMP is the current option, the HIDATE subset selects using the TIMESTMP date. If DEFINED is the current option the DEFINED date is used.

For LISTVTOC the HIDATE and LODATE subset filters entries using the CREATED column date.

For VTOC reports, selection by expiry date is available via the LOEXP/ HIEXP subset.

HIEXP=date  (LISTVTOC SUBSET)

LV  DD=CBLT03   LOEXP=0   HIEXP=+7   * Expire this week

Select files that expire on or before this date.

Notes

Specify a date range by using LOEXP and HIEXP together.

Specifying the same date for LOEXP and HIEXP limits the report to a single expiry date.

The date may be: '+'/-ddd', 'ccymmd', 'ymmd', 'ccyy/mm/dd', 'yy/mm/dd' or a shorter form (Examples are: '+7', '-30', '20020703', '930506', '93/05/06', or '9305', '93/05', or '93').

The century for the 'ymmd' and 'yy/mm/dd' formats of the argument is interpreted using a sliding window in the range 0 to +99 relative to Current Date.

See Year 2000 Compliance for examples of 'Sliding Windows'.
The formats 'dd/mm/yy' and 'mm/dd/yy' are not supported and will result in an ERROR 16.

The format '+'/ddd' allows the date to be expressed as the number of days relative to the current date.

The HIEXP and LOEXP subset filters entries using the EXPIRES column date.

The EXPD/NOEXPD option setting does not affect the selection.

Selection by create date is available via the LODATE/HIDATE subset.

---

**HITRK=nnnn** *(LISTVCAT, LISTVTOC SUBSET)*

**LV DD=DD1 LOCYL=10 HITRK=30**

For CKD Disks only (see HIBLK for FBA). HITRK is used to select files whose extent(s) reside fully or partially below this track limit.

HITRK may be used with LOCYL or LOTRK, but not with HICYL.

See Notes in HIBLK for further information.

---

**HIUSERBA (nn)** *(REPORT VCAT)*

**REPORT VCAT DSN 30 HIUSERBA HIALLRBA FREEBYTES**

Display the High Used Relative Byte Address (for investigative purposes).

The column width is 11, the format '+nnnnnnnnnn' and the heading is HI USE RBA.

**Notes:**

For MVS data sets defined with SMS DATACLASS attribute “Extended Addressability” or VSE VSAM files defined eith IDCAMS attribute EXTRALARGEDATASET, the HIUSERBA value is stored as a relative CI address instead of a relative byte address.

CBLVCAT converts a relative CI address to a relative byte address by multiplying the value by the CISIZE and displays the product in the HIUSERBA column.

The HIUSERBA value is used by CBLVCAT in calculating its tuning recommendations for Catalog reports with TUNE.

---

**IGN=string** *(LISTVCAT, LISTVTOC SUBSET)*

**IGN=/string**

**LC DD=CBLI01 IGN=TEMP**

Ingnore if name starts TEMP.

**LV SYS=10 SUBSET IGN='/A B'**

Ignore if "A B" anywhere in name.

Exclude files from the report based on a reference to the string supplied.

If the string does not start with a slash (i.e. IGN=xxx), the file will be bypassed if the dataset name starts with the string characters.

If the string starts with a slash (i.e. IGN=/xxx), the file will be bypassed if it contains the string characters following the slash anywhere in its name.

**Notes**

If the string contains blank(s) it must be enclosed in quotes (including the slash).

Multiple KEY and/or IGN parameters are allowed and are processed as a logical OR meaning that if KEY and IGN are both specified, the order of appearance is important.

1. **IGN=ONLY KEY=ONL**
   Any file starting with "ONLY" would be ignored as it satisfies the first condition. A file starting with "ONLI" would fail the first condition but be selected by KEY=ONL.

2. **KEY=ONLY IGN=ONLY**
   Any file starting with "ONLY" would be selected by KEY=ONL and included in the report, as would a file starting with "ONLI". The second parameter has no effect as all files starting with "ONLY" have been selected by the first condition.
Display the special index attributes.

The column width is 7, the format is `--IMB--`, `--REP--` or `IMB+REP` and the column heading is `IMB/REP`.

**IMB** is also available within the **KL,RKP/BLK/IMB** combined column in the standard report or within the Combined report field of the same name.

**Notes**

`--IMB--` indicates the file was defined with the **IMBED** attribute. This means that the lowest level of the index (the sequence set) is placed in the first track of each control area it references. The sequence set is automatically replicated on this track.

`--REP--` indicates that the file was defined with the **REPLICATE** attribute. This means that the index CIs are repeated as many times as will fit on a track.

`IMB+REP` indicates that the file was defined with both the **IMBED** and **REPLICATE** attributes.

---

**INDEX**

If **NOINDEX** is the current installation default, **INDEX** can be used to display the index line below the data line for AIX and KSDS files. (**INDEX** and **NOINDEX** are mutually exclusive options).

In the CBL supplied version of **CBLNAME**, **INDEX** is the default (**CBLVCSW3** bit X'20' is off). **NOINDEX** can be made the default by changing this setting.

---

**INFO (nn)**

Display one or more information messages from the available list.

The column width is 21 and the column heading is **INFO** (**INFO** also forms part of the standard report).

The messages which comprise the **INFO** column can be grouped into 2 distinct categories.

**File Messages**, which appear alongside the file to which they relate.

**Summary Messages**, which appear after the last report entry.

The **Messages** which comprise these sections are described as follows:

**File Messages**

**OWNED BY VSAM CATALOG**

The volume is a prime or candidate volume of a VSAM catalog and thus is not available for other VSAM catalogs.

**NOT LAST VOL**

A sequential file has more extents on another volume, or a file has not been closed. Programs that cancel will leave files unclosed, libraries created by programs such as **LIBR** are not closed and Direct Access files are often not closed.

**FILE SERIAL vvvvvv**

A sequential file has more extents on another volume, or a file has not been closed. The serial number is different from the disk volume serial number (For a multi-volume file the serial number shown is that of the first extent). It will occur for all existing files after **MOD NEWVOL**. For **MVS** systems, this field is not always maintained and if the first character is not alphanumeric this field is left blank. Printing of this field can be suppressed (see **CBLVCSW2** in **CBLNAME** and the **CBLVCSW2=X'xx'** run-time override option).

**VOL SEQ nnnnn**

The previous extent number has not been found, indicating that previous extent(s) are on a different volume.

**ERROR IN Fn LABEL**

A problem has been detected in the indicated 'format label'.
***INVALID EXTENT***
The extent is not totally within the disk extent limits held in the Format 4 label in the VTOC.

USER LABEL EXTENT
The extent is a User defined Label Area.

MULTI-EXTENT
The prime extent of a multi-extent file for SORT=EXT reports.

Summary Messages

OLDEST DATED yy/mm/dd
The creation date of the oldest file on the volume (This may be shown as TODAY). The field will not be printed if the SUMMARY option is in force.

LATEST DATED yy/mm/dd
The creation date of the oldest file on the volume (This may be shown as The creation date of the youngest file on the volume (This may be shown as TODAY). The field will not be printed if the SUMMARY option is in force.

VTOC RECS LIVE....nnn
The number of active VTOC entries. (This includes two for the Format 4 label and the Format 5 label). A File may utilise more than one label. The Format 1 label can hold the data for 3 extents. Each additional 13 extents will require a Format 3 label.

VTOC RECS EXPD....nnn
The number of expired VTOC entries (This is independent of the EXPD/ NOEXPD option).

VTOC RECS FREE....nnn
The number of VTOC entries that are totally unused (The number available for use includes the expired count above).

VTOC RECS TOTAL....nnn
The total number of VTOC entries.

vvvvv MOUNTED ON cuu
The device that contains the current volume.

NOALLOC=nn
For VSE systems only, this indicates that some of the files reported are work files which are not allocated until open for output. The numeric value indicates how much space would be used if they were all opened at once (although this is unlikely to happen).

ISC

(LISTVCAT OPTION)

<table>
<thead>
<tr>
<th>DD=BIGCAT   ISC</th>
</tr>
</thead>
<tbody>
<tr>
<td>DD=BIGCAT   ISC  KEY=/NBJ</td>
</tr>
</tbody>
</table>

If NOISC is the current installation default, ISC can be used to invoke CBLVCAT’s In-Storage Catalog processing of VSAM (not ICF) catalogs. (ISC and NOISC are mutually exclusive options).

In the CBL supplied version of CBLNAME, NOISC is the default (CBLNAME option VISCatalog=No so that CBLVCSW9 bit X’04’ is off). ISC can be made the default by changing this setting.

Even if ISC is the installation default, CBLVCAT will not invoke In-Storage Catalog processing when KEY= and/or IGN= subsetting is specified. To override this, for instances when KEY= subsetting results in a significant proportion of a large catalog being selected, explicitly specify the run-time option ISC in the control statements.

Notes:

LISTVCAT operations on large VSAM catalogs can experience unacceptably long execution times. This is a consequence of the VSAM catalog structure. The same is not true of ICF catalogs which have an improved structure which is fully exploited by CBLVCAT. Note that CSA storage for ICF catalog control blocks is requested from above the 16MB line.

VSAM catalogs are designed to optimise direct access to a dataset. The dataset's attributes are held in small (512 byte) control intervals, and each control interval holds information about only 1 dataset.

This design allows CBLVCAT to give good performance when listing small subsets of large catalogs with KEY= and/or IGN= since direct processing can be used to advantage.

However, when listing all datasets (or a large proportion) from a large catalog, direct processing becomes a distinct disadvantage. VSAM will only read multiple control intervals in one physical Input/Output (I/O) operation when using sequential processing. Consequently, using direct processing to report on a large number of datasets requires a large number of physical I/Os, resulting in long execution times.
Using OPTION ISC will dramatically reduce the elapsed time of unacceptably long running jobs. To achieve these gains, CBLVCAT exploits the VSAM Read Ahead Feature to sequentially read the entire Catalog into storage and thereafter uses fast storage access to process records within the Catalog.

The improvements in run-time are subject to the overall system load. For smaller catalogs, in the region of 200 clusters or less, the run-time gains will be less significant.

Where possible, storage for this feature is obtained from above the 16MB line and so users should consider executing CBLVCAT in a VSE partition which allows such storage to be acquired.

**ERROR 042** will be returned almost immediately should insufficient storage be available for the requested catalog to be read. In this event, rerun the job in the largest available batch partition. If enough storage is not available, then use OPTION NOISC to process the catalog.

In summary, the following should be taken into consideration for OPTION ISC processing:

1. The benefits, in terms of elapsed time, will vary with the load on your system and the size of the catalog.
2. Sufficient storage address space is required.
3. Any resulting system contention for virtual storage could impact other batch jobs or on-line response times.
4. SUBSET KEY= and/or IGN= could execute faster without option ISC.

### IXL (nn)    \(\text{(REPORT VCAT)}\)

<table>
<thead>
<tr>
<th>REPORT VCAT</th>
<th>DSN</th>
<th>TYPE</th>
<th>NRECS</th>
<th>IXL</th>
<th>TIMESTMP</th>
</tr>
</thead>
</table>

Display the number of index levels.

- The column width is 3, the format is ‘+nn’ and the column heading is **IXL**.
- This column is also available within the **BUFSP/IXL** combined column in the standard report or within the Combined report field of the same name.

### KEY=string \(\text{(LISTVCAT, LISTVTOC SUBSET)}\)

Select files for the report based on a reference to the string supplied.

- If the string does not start with a slash (i.e. **KEY=xxx**), the file will be included if the dataset name starts with the string supplied.

- If the string starts with a slash (i.e. **KEY=/xxx**), the file will be included if it contains the string characters following the slash **anywhere** in its name.

For MVS systems only, specification of **KEY=string** (but not **KEY=/string**) will automatically generate a **REF=xxx** for the high level qualifier in **string**. As long as an ALIAS exists for that qualifier in the master catalog, the correct user catalog will be searched and so specification of **REF=xxx** is unnecessary.

**Notes**

- If the string contains blank(s) it must be enclosed in quotes (including the slash).
- Multiple **KEY** and/or **IGN** parameters are allowed and are processed as a logical **OR** meaning that if **KEY** and **IGN** are both specified, the order of appearance is important.

1. **IGN=ONLY KEY=ONL**
   - Any file starting with “ONLY” would be ignored as it satisfies the first condition. A file starting with “ONLI” would fail the first condition but be selected by **KEY=ONL**.

2. **KEY=ONL IGN=ONLY**
   - Any file starting with “ONLY” would be selected by **KEY=ONL** and included in the report, as would a file starting with “ONLI”. The second parameter has no effect as all files starting with “ONLY” have been selected by the first condition.
Display the key length for a KSDS/AIX file.

The column width is 3, the format is ‘nnn’ and the column heading is KL.

This column is also available within the KL,RKP/BLK/IMB combined column in the standard report or within the Combined report field parameter KL/BLK/IMB.

The relative key position is available via the RKP report parameter.

Display the key length for a KSDS/AIX file. Display the combined column containing the KSDS/AIX key length and position, VSE SAM blocksize and KSDS imbed/replicate attributes.

The column width is 7, the format is ‘nnn,nnn’ or ‘+nnnnnn’ and the column heading is KL, RKP /BLK/IMB.

The individual columns are available via the KL, RKP BLKSIZE and IMB parameters and KL/BLK/IMB also forms part of the standard report (as shown below).

Example: LISTVCAT KL/BLK/IMB fields (CBL Ref: vmxklkp)

Notes

'nnn,nnn'
Displayed in this format, it is either the Key Length (KL) and the Relative Key Position (RKP) for KSDS files, or, for AIX files, the KL and RKP of the alternate key in the base cluster.

The RKP is the displacement of the key in the record relative to position zero (ie. RKP 10 is the eleventh byte of the record).

'nnnnn'
A display in this format is the file blocksize for SAM (VSE only).

'--IMB--', '--REP--', 'IMB+REP'
Indicates the special index attributes IMBED and/or REPLICATE (See IMB for further details).

A run-time override for the line spacing between files.

The default can be set using the CBLVCSPA (CBL VCat SPAcing) field in CBLNAME, which is supplied containing X'00' (giving a default spacing of 2).

Notes

Specifying LINESPACE=2 is the same as CBLVCSPA=2 or CBLVCSPA=X'02'.
**LIST=YES**

For VSE, **LIST=YES** produces a VTOC report of the drive on which the last successful operation was performed.

**Notes**

Combine this with **EJECT=YES** if the report is required on a new page.

If there was no previous operation a VTOC report of the device assigned to SYS000 will be produced.

**LISTLABL**

**LISTCAT**

**LISTVCAT**

**LISTVTOC**
The LISTVTOC command is used to produce a VTOC report. The output depends upon the OPTION and REPORT commands, together with the SUBSET and MERGE parameters.

VTOC selection is accomplished using the SYS, DEV or VOL parameters (VSE and CMS) or the DDNAME, DEV or VOL parameters (MVS). VSE users may supply a LISTVTOC command with no other parameters or sub-parameters. The result is a VTOC report for the device assigned to SYS000.

If you are a new user please reference Introduction and Guide to List Output for full details and examples of LISTVTOC use.

See Summary of Syntax for a complete list of parameters and sub-parameters available to the LISTVTOC command.

### LMAX (LISTVCAT OPTION)

| LISTVCAT DD=CBLV01 | OPTION | LMAX |

If AVRL (average record length) is the current installation default, LMAX can be used to display the defined maximum record length. AVRL and LMAX are mutually exclusive options.

In the CBL supplied version of CBLNAME, LMAX is the default (CBLVCSW3 bit X'10' is off), however AVRL may be made the default by changing this switch.

Note

For notes relating to LMAX see REPORT VCAT LMAX below.

### LMAX (nnn) (REPORT VCAT)

| REPORT VCAT DSN | TYPE NRECS LMAX AVRL TIMESTAMP |

Display the maximum record length as specified at define time.

The column width is 7, the format is ‘+nnnnnnV’ and the heading is LMAX (LMAX forms part of the standard Catalog report if the LMAX / AVRL option is set to LMAX).

Notes

For RRDS files, this is the actual length, but for other types of file, the true length may be any value up to LMAX.

A "V" appended to the right of the numerical value (nnnnnV), indicates that the file's record lengths are variable.

For a non-loaded file (one that has been defined but has had no data loaded) this means that it was defined with an average and maximum record length that differ. On loaded files, it can also mean that the defined length is inconsistent with the space the file occupies (it is possible to define the record length as 2000 and then only load 30 byte records!). Tuning calculations will, in this case, be based on the effective record length.

Files defined with maximum and average record length equal, but which have records of differing sizes loaded, will also be suffixed with the 'V'.

Files defined with maximum and average record length different, but which have records of only equal sizes loaded, will not be suffixed with the 'V'.

### LOBLK=nnnn (LISTVCAT, LISTVTOC SUBSET)

| LV VOL=CBLT06 | LOBLK=13000 | HIBLK=13999 @h0 |

For FBA Disks only (see HICYL or HITRK for CKD). LOBLK is used to select files whose extent(s) reside fully or partially above this block limit.

See Notes in HIBLK for further information and rules regarding the display of extent information.
**LOCISZ=nnnn**

(LISTVCAT SUBSET)

LISTVCAT DD=CBLI11   LOCISZ=2048   HICISZ=4096

Select files with a CI size greater than or equal to the value specified.

**LOCISZ** may be used with **HICISZ** to restrict to a specified range of CI sizes.

---

**LOCYL=nnnn**

(LISTVCAT, LISTVTOC SUBSET)

LV DD=DD1   SORT=DSN   LOCYL=2   HICYL=10
LC DD=CBLV05   LOCYL=882   HICYL=885 @h0

For **CKD Disks only** (see **LOBLK** for FBA). **LOCYL** is used to select files whose extent(s) reside fully or partially above this cylinder limit.

See Notes in **HIBLK** for further information.

---

**LODATE=date**

(LISTVCAT, LISTVTOC SUBSET)

LV VOL=CB9009   LODATE=930614   HIDATE=93/10/12
LC DD=CBLV05   LODATE=-15   * Files Changed in the last 15 days

Select files with a date greater than or equal to the value specified.

**Notes**

For notes on use and format of **LODATE** arguments, see **HIDATE**.

---

**LOEXP=date**

(LISTVTOC SUBSET)

LV DD=CBLT03   LOEXP=93/10/27
LV DD=CBLT03   LOEXP=0   HIEXP=+1   * Expire today or tommorrow

Select files that expire on or after this date.

Specify a date range by using **HIEXP** and **LOEXP** together.

For notes on use and format of **LOEXP** arguments, see **HIEXP**.

---

**LOTRK=nnnn**

(LISTVCAT, LISTVTOC SUBSET)

LISTVTOC DDNAME=DD1   LOTRK=10   HITRK=30

For **CKD Disks only** (see **LOBLK** for FBA). **LOTRK** is used to select files whose extent(s) reside fully or partially above this track limit.

**LOTRK** may be used with **HITRK** to define upper and lower boundaries.

See Notes in **HIBLK** for further information and rules regarding the display of extent information.

---

**LRECL (nn)**

(REPORT VTOC)

REPORT VTOC   DSN TYPE BLKSIZE LRECL RECFM KEYLP

Display the logical record length.

The column width is 6, the format is ‘+nnnnn’ and the heading is **LRECL** (LRECL also forms part of the standard report).
VSE SAM does not automatically record a file's geometry (RECFM, LRECL and BLKSIZE) in disk VTOCs. However, it is common for VSE applications that perform I/O to include this information in the DTF control block and so write the file's geometry to its Format 1 record in the VTOC. CBLVCAT reports the lrecl value, if present, otherwise blanks are displayed in the LRECL column.

**MAXLRECL=nnnn/KEEP**

Sets the maximum record length for a tuning run.

It is provided as an override to the the maximum record length which was set at DEFINE time (see `LMAX`) `MAXLRECL=KEEP` uses the defined value rather than allowing CBLVCAT to increase it.

See `AVLRECL=nnn` for a discussion of `AVLRECL` and `MAXLRECL` for AIX files and the effect of `AVLRECL` on maximum record length.

---

**MERGE**

MERGE may be used on a `LISTVTOC/ LISTVCAT` command provided a `REPORT` statement for the appropriate operation has already been supplied. It causes the output from the current command to be merged with the output from the command that follows, enabling the listings from two or more commands to be combined into one report (See examples 12-18 in Guide to List Output).

Notes

If no `SORT` or `SORTD` parameter is supplied, MERGE will sort on the first `REPORT` parameter.

Using MERGE results in 'DATASET NAME' being used as the heading for the `DSN` column, if specified.

If the applicable `REPORT` statement does not include a `SORT` parameter, a default sort (ascending) is performed with the first parameter in the `REPORT` list as the primary sort field.

A MERGE of several commands referencing the same Catalog/VTOC with different `SUBSET` parameters is effectively an OR condition. Any entries which would be selected by multiple `SUBSET` parameters will only be reported once (see example 13 in Guide to List Output).

WARN 010 messages (NO FILES SELECTED) are suppressed when merging `SUBSET` output from more than one report.

For mixed reports (`LISTVCAT` with `LISTVTOC`), the report headings and sort positions are taken from the `REPORT` parameters applicable to the last command in the MERGE list. It is therefore possible to have `LISTVTOC` headings over unrelated `LISTVCAT` data and vice-versa.

For `LISTVTOC` For `DEV=ALL MERGE` (or `LISTVCAT` (or `REF=ALL MERGE` - ICF with DFP 3.1 or higher) the multiple `LISTVTOC` (or Catalog) listings are combined into one customised report. If either of these reports need to be merged with another report a second `MERGE` parameter is required.

e.g.

```
LV DEV=ALL MERGE
LC REF=ALL MERGE
```

**ERROR 16** is issued for MERGE if a `REPORT` command is not supplied prior to the relevant `LISTVCAT/LISTVTOC` (or set of `LISTVCAT/LISTVTOC`) command(s).

---

**MOD**

For VSE non-VSAM managed files only, allows modification of a file's data set name, expiry date, volume serial number and owner-id.
Note

Select and optionally check a device using one, or any combination of the DEV, DEV, SYS and VOL parameters. The OWN parameter may also be used as a means of checking the device.

For modification of a disk's volume serial number, VOL is used only as a check that the correct disk has been selected. i.e. a DEV and/or SYS and DEV and/or SYS parameter must also be coded.

WARNING: Take great care when modifying disk volume serial numbers.

---

**MOUNT**

**LISTVCAT OPTION**

<table>
<thead>
<tr>
<th>LC DD=ICF01 OPTION MOUNT</th>
</tr>
</thead>
</table>

For ICF Catalogs only. If NOMOUNT is the current installation default, MOUNT can be used to cause a LISTVCAT command to issue system mount requests for any VVDS volume not on line (MOUNT and NOMOUNT are mutually exclusive options).

In the CBL supplied version of CBLNAME, MOUNT is the default (CBLVCSW3 X'80' bit off). NOMOUNT may be made the default by changing this switch.

---

**NEWDSN=xxx.xxx.xx.etc**

**(MOD)**

<table>
<thead>
<tr>
<th>MOD DEV=161 VOL=PACK11 DSN='FILE ABC' NEWDSN='FILE.ABC'</th>
</tr>
</thead>
</table>

* Choose device
* Check volume (optional)
* Quotes required (for blanks)
* No quotes required

For VSE systems only, change a file's data set name.

Notes

If the new name contains blanks, the NEWDSN argument must be enclosed in single quotes.

The file's current data set name must be supplied via the DSN parameter.

Select and optionally check a device using one, or any combination of the DEV, SYS and VOL parameters. The OWN parameter may also be used as a means of checking the device.

---

**NEWOWN=xxx**

**(MOD)**

<table>
<thead>
<tr>
<th>MOD SYS=1 OWN=PROJECT1 NEWOWN=PROJ2</th>
</tr>
</thead>
</table>

* Choose device
* Check owner (optional)
* Up to 14 chars

For VSE systems only, used to modify the 14-byte owner field in the disk volume label.

One to fourteen alphanumeric characters must be provided (if less than 14 bytes are provided, the user field is left justified and padded with blanks).

Notes

If the owner-id contains blanks, the NEWOWN argument must be enclosed in single quotes.

Select and optionally check a device using one, or any combination of the DEV, SYS and VOL parameters. The OWN parameter may also be used as a means of checking the device.

---

**NEWVOL=xxxxxx**

**(MOD)**

<table>
<thead>
<tr>
<th>MOD SYS=1 VOL=PACK11 NEWVOL=PACK12</th>
</tr>
</thead>
</table>

* Choose device
* Check volume (optional)
* New volume serial number

For VSE systems only, used to change the disk volume serial number.

Six alphanumeric digits must be provided.
Select and optionally check a device using one, or any combination of the **DEV**, **SYS** and **VOL** parameters. The **OWN** parameter may also be used as a means of checking the device.

**NOALIAS**

<table>
<thead>
<tr>
<th>LC</th>
<th>DD=CBLI11</th>
<th>OPTION</th>
<th>NOALIAS</th>
</tr>
</thead>
</table>

For **MVS** systems only. If **ALIAS** is the current installation default, **NOALIAS** can be used to stop the alias items appearing as separate entries in the report (They will still appear as a reference for the appropriate **USERCAT**). **ALIAS** and **NOALIAS** are mutually exclusive **OPTIONs**.

In the CBL supplied version of **CBLNAME**, **ALIAS** is the default (**CBLVCSW1** bit X'08' is on). **NOALIAS** can be made the default by changing this setting.

**NOASSOC**

<table>
<thead>
<tr>
<th>LC</th>
<th>DD=CBLV01</th>
<th>OPTION</th>
<th>NOASSOC</th>
</tr>
</thead>
</table>

If **ASSOC** is the current installation default, **NOASSOC** can be used to suppress file association information on the Catalog report (**ASSOC** and **NOASSOC** are mutually exclusive options).

In the CBL supplied version of **CBLNAME**, **ASSOC** is the default (**CBLVCSW1** bit X'80' is on). **NOASSOC** can be made default by changing this switch.

See **ASSOC** for speed considerations when using a **KEY** parameter.

**NODEFINE**

<table>
<thead>
<tr>
<th>LC</th>
<th>DD=CBLV01</th>
<th>OPTION</th>
<th>NODEFINE</th>
</tr>
</thead>
</table>

If **DEFINE** is the current installation default, **NODEFINE** can be used to suppress the output file of IDCAMS **DELETE/DEFINE** statements. (**DEFINE** and **NODEFINE** are mutually exclusive options).

In the CBL supplied version of **CBLNAME**, **NODEFINE** is the default (**CBLVCSW6** X'80' bit off). **DEFINE** may be made the default by changing this switch.

**NOEXPD**

| LV   | VOL=CBLT01 KEY=/CICS | OPTION | NOEXPD | NOPERM |

If **EXPD** is the current installation default, **NOEXPD** can be used to display the expiry date (instead of 'EXPD') for files that have expired (**EXPD** and **NOEXPD** are mutually exclusive options).

In the CBL supplied version of **CBLNAME**, **EXPD** is the default (**CBLVCSW2** bit X'04' is off). **NOEXPD** can be made the default by changing this setting.

See **EXPD** for an example showing the effect of **NOEXPD** on the Freespace table.

**NOFREE**

<table>
<thead>
<tr>
<th>LV</th>
<th>VOL=CBLT01</th>
<th>OPTION</th>
<th>NOFREE</th>
</tr>
</thead>
</table>

If **FREETAB** (or **FREE**) is the current installation default, **NOFREE** can be used to suppress Freespace extent information. (**FREE**, **FREETAB** and **NOFREE** are mutually exclusive options).

In the CBL supplied version of **CBLNAME**, **NOFREE** is the default (**CBLVCSW2** bit X'10' is off). **FREE** or **FREETAB** can be made the default by changing this setting.
NOGDGDSN  
(ListVcat Option)  

LC DD=CBLI22 NOGDGDSN NOGDGRPT * Override CBLVCWS6=X'60' (X'40'+X'20')

If GDGDSN is the current installation default, NOGDGDSN can be used to suppress the GDG Dataset Name (GDGDSN and NOGDGDSN are mutually exclusive options).

In the CBL supplied version of CBLNAME, NOGDGDSN is the default (CBLVCWS6 bit X'20' set off). GDGDSN can be made the default by changing this setting.

NOGDGRPT  
(ListVcat Option)  

LC DD=CBLI22 NOGDGDSN NOGDGRPT * Override CBLVCWS6=X'60'

If GDGRPT is the current installation default, NOGDGRPT can be used to suppress the repetition of unchanged GDG information on overflow information lines (GDGRPT and NOGDGRPT are are mutually exclusive options).

See GDGDSN for an example.

In the CBL supplied version of CBLNAME, NOGDGRPT is the default (CBLVCWS6 bit X'40' set off). GDGRPT can be made the default by changing this setting.

NOINDEX  
(ListVtoc Option)  

LC DD=CBLI22 OPTION NOINDEX * Override CBLVCWS3=X'20'

If INDEX is the current installation default, NOINDEX can be used to suppress index lines for AIX and KSDS files (INDEX and NOINDEX are mutually exclusive options).

In the CBL supplied version of CBLNAME, INDEX is the default (CBLVCWS3 bit X'20' is off). NOINDEX can be made the default by changing this setting.

Note

NOINDEX can be useful for reports that will be post-processed as output will be one line per file.

NOISC  
(ListVcat Option)  

LC DD=VSESPUC NOISC * Suppress In-storage Catalog processing.

If ISC is the current installation default, NOISC can be used to suppress CBLVCAT's In-Storage Catalog processing of VSAM catalogs. (ISC and NOISC are mutually exclusive options).

In the CBL supplied version of CBLNAME, NOISC is the default (CBLNAME option VISCatalog=No so that CBLVCWS9 bit X'04' is off).

See option ISC for notes on In-Storage Catalog processing.

NOMOUNT  
(ListVcat Option)  

LC DD=ICF01 OPTION NOMOUNT * Override CBLVCWS3=X'80'

For ICF Catalogs only. If MOUNT is the current installation default, NOMOUNT can be used to suppress system mount requests for any VVDS volume not on line (MOUNT and NOMOUNT are mutually exclusive options).

In the CBL supplied version of CBLNAME, MOUNT is the default (CBLVCWS3 X'80' bit off). NOMOUNT may be made the default by changing this switch.

Note

Any offline VVDS volume encountered will cause a VVDS NOT AVAILABLE warning message.
NOOVLAY  

**NOOVLAY**

**(LISTVCAT,LISTVTOC OPTION)**

For the standard report only. If OVLAY is the current installation default, NOOVLAY can be used to suppress the overwrite of fields following a data set name which is longer than the DSN field. The data set name name is printed immediately and all other fields are printed on the following line. (OVLAY and NOOVLAY are are mutually exclusive options).

In the CBL supplied version of CBLNAME, NOOVLAY is the default (CBLVCSW3 bit X'40' set off). OVLAY can be made the default by changing this setting.

NOPCAP  

**NOPCAP**

**(LISTVCAT OPTION)**

If PRTCAP is the current installation default, NOPCAP can be used to suppress print of the third tuning output block (the CAP Block). PRTCAP and NOPCAP are are mutually exclusive options.

In the CBL supplied version of CBLNAME, PRTCAP is the default (CBLVCSW3 bit X'04' set off). NOPCAP can be made the default by changing this setting.

See CAPacity block in Tuning Output for details of this tuning block.

NOPERM  

**NOPERM**

**(LISTVTOC OPTION)**

If PERM is the current installation default, NOPERM can be used to display the expiry date (instead of 'PERMANENT') for files that expire on or after 99/365 (PERM and NOPERM are mutually exclusive options).

In the CBL supplied version of CBLNAME, PERM is the default (CBLVCSW2 bit X'02' is off). NOPERM can be made the default by changing this setting.

Note

Dates will be displayed as ‘yy/mm/dd’ or ‘20yy/mm/dd’.

NOPJCL  

**NOPJCL**

**(LISTVCAT OPTION)**

If PRTJCL is the current installation default, NOPJCL can be used to suppress print of the fourth tuning output block (the JCL Block). PRTJCL and NOPJCL are are mutually exclusive options.

In the CBL supplied version of CBLNAME, PRTJCL is the default (CBLVCSW3 bit X'02' set off). NOPJCL can be made the default by changing this setting.

See JCL block in Tuning Output for details of this tuning block.

NOPSEV  

**NOPSEV**

**(LISTVCAT OPTION)**

If PRTSEV is the current installation default, NOPSEV can be used to suppress print of the first tuning output block (the SEV Block). PRTSEV and NOPSEV are are mutually exclusive options.

In the CBL supplied version of CBLNAME, PRTSEV is the default (CBLVCSW3 bit X'08' set off). NOPSEV can be made the default by changing this setting.

See SEV block in Tuning Output for details of this tuning block.
NOVOL (LISTVCAT OPTION)

If VOLINFO is the current installation default, NOVOL can be used to suppress the volume information (VOLINFO and NOVOL are mutually exclusive options).

In the CBL supplied version of CBLNAME, VOLINFO is the default (CBLVCSW1 bit X'10' set on). NOVOL can be made the default by changing this setting.

See JCL block in Tuning Output for details of this tuning block.

Note

Specifying CBLVCEXT=nn will force VOLINFO.

NRECS (nn) (REPORT VCAT)

Display the number of records in each file.

The column width is 10, the format is ‘+nnnnnnnn+’ and the heading is NRECS. (NRECS also forms part of the Standard report).

Notes

The value displayed is the total number of records loaded (except for RRDS files when it is the number of records inserted).

The value displayed will not be accurate if the file is currently in use by another program, as the catalog is not updated until file closure.

A ‘+’ suffix (nnnnnn+) indicates that the Catalog record count has not been maintained by VSAM (This generally occurs when processing has been by control interval access instead of by logical record). In this case the value displayed is CBLVCAT’s estimate based upon the high used RBA. (The estimate assumes that all records are of the defined maximum length and that the last control area contains one record, i.e. the minimum number of records).

An empty file (one that has been defined but not loaded) is displayed in one of the following formats: 0(nnnn), 0(nn.nK), 0(nnnK), 0(nn.nM), 0(nnnM).

K or M indicate KILO or MEGA and the value in parentheses indicates the number of records that can be loaded into the primary allocation (See ALLOCP).

NRECS=nnnn (LISTVCAT SUBSET)

Select files containing a minimum of nnnn records.

When specified as NRECS=0 the subset selects all empty files.

NSEC (nn) (REPORT VCAT)

Display the number of secondary extents.

The column width is 4, the format is ‘+nnn’ and the heading is NSEC.

Notes

NSEC reports the number of extents, not allocations, since it is the number of extents which limits the expansion of the file.

Each allocation, Primary and Secondary, may be satisfied with 1 or more extent(s). Therefore, secondary extents may exist for a file that has a single, primary allocation.
ICF and VSAM support up to 123 extents, except for VSAM cataloged REUSE or UNIQUE files which are restricted to 16 extents per volume. NSEC assists the user in identifying fragmentation, i.e. more than 1 extent, of Primary and Secondary Allocations.

Severity messages are generated when the number of secondary extents reaches the CBLVCALW and CBLVCALE secondary extent thresholds. See CBLNAME for the threshold values and Messages for an expansion of the messages.

Additional secondary extent warnings are shown as part of the ALLOCS field (The extents themselves can also be shown depending upon the CBLVCEXT CBLNAME switch or CBLVCEXT=nn run-time option).

See note in ALLOCS regarding Secondary Extents and Allocations.

**OPTION**

(OPTION, LISTVCAT, LISTVTOC)

**OPTIONS**

OPT

**OPTION** RECFM FREE NOEXPD
LV VOL=CBLT04

**OPTION** ALIAS NOASSOC OVLAY
LC DD=CBLV91

The keyword **OPTION** indicates that report tailoring sub-parameters follow.

The sub-parameters select the report fields (Standard report) and also the type and amount of data to be printed for each file (Standard and Customised reports).

**Notes**

**OPTION** may be used as an operation itself, or as a parameter of another operation word. When used as a parameter, the word **OPTION** is optional as all its associated sub-parameters are unique.

Options are set for the remainder of the run, or until reset by another **OPTION** parameter.

Any **LISTVTOC** option may be specified during a **LISTVCAT** operation and vice-versa.

All **OPTION** sub-parameters are part of CBLNAME and can be made the installation default (see CBLNAME).

See **OPTION Summary** for a full list of **OPTION** parameters.

**OVLAY**

(LISTVCAT, LISTVTOC OPTION)

**LC** DD=CBLI22 OVLAY

For the **Standard report only**. If **NOOVLAY** is the current installation default, **OVLAY** can be used to allow overwrite of the field(s) that follow DSN, by data set names that are longer than the DSN field width. (**FREE**, **FREETAB** and **NOFREE** are mutually exclusive options).

In the CBL supplied version of CBLNAME, **NOOVLAY** is the default (CBLVCSW3 bit X'40' is off). **OVLAY** can be made the default by changing this setting.

**OWN=xxx**

(DEL, LISTVTOC, MOD)

**OWNER=xxx**

MOD SYS=1
OWN=PROJECT1
NEWOWN=PROJECT

* Choose device
* Check owner (optional)
* Up to 14 chars

For **VSE** systems only, the **OWN** parameter used to check the volume owner-id of the selected volume.

**Notes**

If **OWN** does not match the owner-id, further processing depends on the **FAIL** parameter. (which defaults to **CANCEL**).

Select the volume via the **DEV** and/or **SYS** parameters.

The **owner field** in the **VTOC** may be modified via **MOD NEWOWN=xxx**.
**PAGEDEPTH=nn**  
(ListVCAT, LISTVTOC option)

PD=nn

| LC | DD=CBLI11 | PAGEDEPTH=32767 | * One set of headings. |

A run-time override for the number of lines on the page.

The default can be set using the **CBLCLINE** (CBL Common LINEs) field in **CBLNAME**, which is supplied containing X'00' (resulting in 58 lines for MVS and the SYSLST system default for VSE). Take care when changing the **CBLNAME** value **CBLCLINE** as **SELCOPY** users will be affected.

Notes
Setting a very high **PAGEDEPTH**, e.g. 32767 (32K-1), causes no page throws and consequently only one set of headings. This can be a useful technique when post-processing reports.

---

**PAGEWIDTH=nn**  
(ListVCAT, LISTVTOC option)

PW=nn

| OPTION | PAGEWIDTH=80 | * Gives PW 80 |
| LC | DD=CBLI01 | PAGEWIDTH=160 | * Gives PW 160 |
| REPORT VTOC | DSN CREATED | * Default to PW 72 |

A run-time override for the report page width.

Heading and footing lines are adjusted according to the **PAGEWIDTH** parameter.

The default can be set using the **CBLCPW** (CBL Common Page Width) field in **CBLNAME**, which is supplied containing AL1(0) (See notes below). Take care when changing the **CBLNAME** value **CBLCPW** as **SELCOPY** users will be affected.

Notes
Data within the report lines are unaffected by this parameter.

The argument of the last **PAGEWIDTH** parameter takes effect over the entire report.

The minimum **PAGEWIDTH** is 72 bytes and the maximum is 160 bytes (ERROR 061 - INVALID PAGE GEOMETRY will be generated for an invalid **PAGEWIDTH**).

If **PAGEWIDTH** is not coded and the **CBLNAME** value for **CBLCPW** is set to zero (AL1(0)), **CBLVCAT** selects an appropriate heading width. For a standard report this is 132 characters, whereas for **REPORT VCAT/VTOC** the width as implicitly defined by the **REPORT** operation will be used.

The Page Number value is always right adjusted to the defined or implied **PAGEWIDTH**. The Operating System and Jobname are omitted and the date in the heading is compressed into the International Date Standard Format (YY/MM/DD HH:MM).

The time and date will be shortened or removed altogether for small page widths with large headings.

---

**PASS=passwd**  
(ListVCAT)

| LC | DD=CBLI11 | PASS=CBL |

Specify the VSAM read password for the catalog (if one is required).

When a password is in effect and none is specified, the system, not **CBLVCAT**, will ask the operator to supply it.

---

**PCNT**  
(ListVCAT option)

| LC | DD=CBLI11 | OPTION | PCNT | * Override CBLVCSW1=X'00' |

If **UNUSED** is the current installation default, **PCNT** can be used to display the percentage of space used (**PCNT**) column instead of the amount of space unused column (**UNUSED**). (**PCNT** and **UNUSED** are mutually exclusive options).
In the CBL supplied version of CBLNAME, PCNT is the default (CBLVCSW1 bit X'04' is on). UNUSED can be made default by changing this switch.

See the following REPORT VCAT PCNT for further information and ALLOC3 for a description of the other fields generated by this option.

---

**PCNT (nn)**

**REPORT VCAT**  
**DSN 22**  
**TYPE NRECS**  
**PCNT**  
**ALLOC4**

Display the percentage of currently allocated space that contains data.

The column width is 8, the format is "**nn.n**" and the heading is PCNT (PCNT also forms part of the standard report if PCNT/UNUSED option is set to PCNT).

**Notes**

PCNT is calculated as the ratio of the current record count (the NRECS field) compared to CBLVCAT's estimate of the capacity of the total allocation (the ALLOCT field).

If the PCNT value has reduced even though the number of records has increased, the allocation size has changed. This will be shown if you display the allocation columns.

"**nn.n**"  
Indicates that the percent full warning threshold has been reached (This also enables the SEV 1-09 warning message). This default threshold may be changed via CBLVCPCF in CBLNAME, or at run time via the CBLVCPCF option.

"**ALL **"  
The space currently allocated is full and additions to the file will cause secondary allocations.

"**FULL**"  
The space currently allocated is full and no secondary allocation has been specified (The file cannot be expanded).

---

**PERM**

**LISTVTOC OPTION**

**LV**  
**VOL=CBLT01**  
**PERM**  
**EXPD**

If NOPERM is the current installation default, PERM can be used to display 'PERMANENT' (instead of the actual expiry date) for files that have an explicit expiry date of 1999/365 or 1999/366. (PERM and NOPERM are mutually exclusive options).

Note that an expiry date of 1999/12/31 which has been calculated from a retention period value is not considered permanent by the operating system.

In the CBL supplied version of CBLNAME, PERM is the default (CBLVCSW2 bit X'02' is off). NOPERM can be made the default by changing this setting.

---

**PHYREC (nn)**

**REPORT VCAT**  
**DSN 22**  
**NRECS**  
**PHYREC**

Display the Physical Record size used by VSAM.

The column width is 7, the format is '+nnnnn' and the heading is PHYREC.

**Notes**

The range of allowable physical record sizes depends on the operating system. The actual value that VSAM chooses depends on the VSAM level, the file's CI size and the device type.

For tuning recommendations, CBLVCAT will choose a CI size based on the appropriate physical record size (i.e. the CI size is chosen so that 1 CI contains 1 physical record which results in minimum I/O).
**PRTCAP** *(LISTVCAT OPTION)*

If NOPCAP is the current installation default, PRTCAP can be used to display the third tuning output block (the CAP Block). PRTCAP and NOPCAP are mutually exclusive options.

In the CBL supplied version of CBLNAME, PRTCAP is the default (CBLVCSW3 bit X’04’ is off). NOPCAP can be made default by changing this switch.

See Tuning Output for details of the CAPacity Block.

---

**PRTJCL** *(LISTVCAT OPTION)*

If NOPJCL is the current installation default, PRTJCL can be used to display the fourth tuning output block (the JCL Block). PRTJCL and NOPJCL are mutually exclusive options.

In the CBL supplied version of CBLNAME, PRTJCL is the default (CBLVCSW3 bit X’02’ is off). NOPJCL can be made default by changing this switch.

See Tuning Output for details of the JCL Block.

---

**PRTSEV** *(LISTVCAT OPTION)*

If NOPSEV is the current installation default, PRTSEV can be used to display the first tuning output block (the SEV Block). PRTSEV and NOPSEV are mutually exclusive options.

In the CBL supplied version of CBLNAME, PRTSEV is the default (CBLVCSW3 bit X’08’ is off). NOPSEV can be made default by changing this switch.

See Tuning Output for details of the SEVerity Block.

---

**QUERY** *(COMMAND)*

The QUERY command can only be used with the CBLNAME parameter and displays the contents of the CBLVCAT fields within CBLNAME (see QUERY CNLNAME in the chapter CBLNAME for an example).

**Notes**

Any temporary changes (via OPTION parameters) will be reflected in the current values displayed.

The QUERY command, unlike other operations, must be complete on one logical record.

---

**RAW=fnme** *(LISTVCAT,LISTVTOC OPTION)*

RAW=fnme may be specified as an option on LISTVCAT, LISTVTOC or LISTLABL in order to write a record containing all available REPORT fields to the specified file for each dataset listed.

fnme is the MVS ddname or VSE label assigned to a pre-allocated data set.
This allows users to generate output which may easily be postprocessed using a utility such as SELCOPY.

The format of each record is as follows:

**LISTVCAT Fields** (Max LRECL 542)

<table>
<thead>
<tr>
<th>REPORT field</th>
<th>Offset</th>
<th>Length</th>
<th>REPORT field</th>
<th>Offset</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSN</td>
<td>000</td>
<td>44</td>
<td>CI/CA</td>
<td>248</td>
<td>06</td>
</tr>
<tr>
<td>TYPE</td>
<td>044</td>
<td>08</td>
<td>SHR</td>
<td>254</td>
<td>05</td>
</tr>
<tr>
<td>NRECS</td>
<td>052</td>
<td>10</td>
<td>S/C</td>
<td>259</td>
<td>05</td>
</tr>
<tr>
<td>PCNT</td>
<td>062</td>
<td>08</td>
<td>DEFINED</td>
<td>264</td>
<td>11</td>
</tr>
<tr>
<td>ALLOCCT</td>
<td>070</td>
<td>08</td>
<td>EXPIRES</td>
<td>284</td>
<td>10</td>
</tr>
<tr>
<td>ALLOCP</td>
<td>086</td>
<td>08</td>
<td>SPLITC1</td>
<td>294</td>
<td>05</td>
</tr>
<tr>
<td>ALLOCS</td>
<td>094</td>
<td>10</td>
<td>SPLITCA</td>
<td>299</td>
<td>05</td>
</tr>
<tr>
<td>FRSP</td>
<td>104</td>
<td>07</td>
<td>SEVL</td>
<td>304</td>
<td>04</td>
</tr>
<tr>
<td>LMAX</td>
<td>111</td>
<td>07</td>
<td>VOLUME</td>
<td>308</td>
<td>06</td>
</tr>
<tr>
<td>KL/BLK/IMB</td>
<td>118</td>
<td>08</td>
<td>GMAX</td>
<td>314</td>
<td>04</td>
</tr>
<tr>
<td>CISIZE</td>
<td>126</td>
<td>07</td>
<td>GVER</td>
<td>318</td>
<td>04</td>
</tr>
<tr>
<td>BUFSF/IXL</td>
<td>133</td>
<td>07</td>
<td>GGEN</td>
<td>323</td>
<td>05</td>
</tr>
<tr>
<td>EXCPS</td>
<td>140</td>
<td>07</td>
<td>STD1</td>
<td>329</td>
<td>10</td>
</tr>
<tr>
<td>TIMESTMP</td>
<td>147</td>
<td>20</td>
<td>STD2</td>
<td>339</td>
<td>08</td>
</tr>
<tr>
<td>ALLOCU</td>
<td>167</td>
<td>06</td>
<td>HIUSERBA</td>
<td>347</td>
<td>11</td>
</tr>
<tr>
<td>NSEC</td>
<td>173</td>
<td>04</td>
<td>HIALLRA</td>
<td>358</td>
<td>11</td>
</tr>
<tr>
<td>AVRL</td>
<td>178</td>
<td>07</td>
<td>FREEBYTES</td>
<td>369</td>
<td>11</td>
</tr>
<tr>
<td>PHYREC</td>
<td>185</td>
<td>07</td>
<td>COMPONENT</td>
<td>380</td>
<td>44</td>
</tr>
<tr>
<td>RECSTATS</td>
<td>192</td>
<td>24</td>
<td>ENTRY</td>
<td>424</td>
<td>44</td>
</tr>
<tr>
<td>KL</td>
<td>216</td>
<td>03</td>
<td>SMSS</td>
<td>468</td>
<td>08</td>
</tr>
<tr>
<td>RKP</td>
<td>219</td>
<td>05</td>
<td>SMSD</td>
<td>477</td>
<td>08</td>
</tr>
<tr>
<td>BLKSIZE</td>
<td>224</td>
<td>07</td>
<td>SMSM</td>
<td>486</td>
<td>08</td>
</tr>
<tr>
<td>IMB/REP</td>
<td>231</td>
<td>07</td>
<td>EXT</td>
<td>494</td>
<td>04</td>
</tr>
<tr>
<td>BUFSF</td>
<td>238</td>
<td>07</td>
<td>CATALOG</td>
<td>498</td>
<td>44</td>
</tr>
<tr>
<td>IXL</td>
<td>245</td>
<td>03</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**LISTVTORC Fields** (Max LRECL 175)

<table>
<thead>
<tr>
<th>REPORT field</th>
<th>Offset</th>
<th>Length</th>
<th>REPORT field</th>
<th>Offset</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSN</td>
<td>000</td>
<td>44</td>
<td>BLKSIZE</td>
<td>106</td>
<td>07</td>
</tr>
<tr>
<td>CYL/HD</td>
<td>044</td>
<td>16</td>
<td>LRECL</td>
<td>113</td>
<td>06</td>
</tr>
<tr>
<td>CISIZE</td>
<td>060</td>
<td>06</td>
<td>RECFM</td>
<td>119</td>
<td>05</td>
</tr>
<tr>
<td>START</td>
<td>066</td>
<td>08</td>
<td>CREATED</td>
<td>124</td>
<td>10</td>
</tr>
<tr>
<td>ALLOC</td>
<td>074</td>
<td>07</td>
<td>INFO</td>
<td>134</td>
<td>21</td>
</tr>
<tr>
<td>USED</td>
<td>081</td>
<td>07</td>
<td>VOLUME</td>
<td>155</td>
<td>06</td>
</tr>
<tr>
<td>TYPE</td>
<td>088</td>
<td>07</td>
<td>ACCESSED</td>
<td>161</td>
<td>10</td>
</tr>
<tr>
<td>EXPIRES</td>
<td>095</td>
<td>11</td>
<td>UNIT</td>
<td>171</td>
<td>04</td>
</tr>
</tbody>
</table>

**LISTLABL Fields** (Max LRECL 136)

<table>
<thead>
<tr>
<th>REPORT field</th>
<th>Offset</th>
<th>Length</th>
<th>REPORT field</th>
<th>Offset</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>PART</td>
<td>000</td>
<td>05</td>
<td>EXPY</td>
<td>083</td>
<td>08</td>
</tr>
<tr>
<td>FNAM</td>
<td>005</td>
<td>07</td>
<td>BUFI</td>
<td>091</td>
<td>05</td>
</tr>
<tr>
<td>NAME</td>
<td>012</td>
<td>44</td>
<td>BUFD</td>
<td>096</td>
<td>05</td>
</tr>
<tr>
<td>UNIT</td>
<td>057</td>
<td>07</td>
<td>STRT</td>
<td>101</td>
<td>11</td>
</tr>
<tr>
<td>VSER</td>
<td>064</td>
<td>06</td>
<td>ALLC</td>
<td>112</td>
<td>11</td>
</tr>
<tr>
<td>CATY</td>
<td>070</td>
<td>07</td>
<td>RECL</td>
<td>124</td>
<td>05</td>
</tr>
</tbody>
</table>
**RECDEL (nn)**

Display the number of records deleted since the file was defined or re-loaded.

The column width is 6, the format is '+nnnnK' and the heading is **RECS DELETD** (**RECS DELETD** also forms part of the **RECSTATS** combined column).

**Notes**

The file statistics are not always reliable, particularly if the file has been updated using control interval access.

---

**RECFM (nn)**

Display the Record Format.

The column width is 5 and the column heading is **RECFM** (**RECFM** also forms part of the standard report).

The possible values are:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASCII</td>
<td>The file is encoded in ASCII, not EBCDIC.</td>
</tr>
<tr>
<td>F</td>
<td>Fixed, unblocked.</td>
</tr>
<tr>
<td>FA</td>
<td>Fixed, unblocked, ASA characters.</td>
</tr>
<tr>
<td>FB</td>
<td>Fixed, Blocked.</td>
</tr>
<tr>
<td>FBA</td>
<td>Fixed, Blocked, ASA characters.</td>
</tr>
<tr>
<td>FBS</td>
<td>Fixed, Standard Blocks.</td>
</tr>
<tr>
<td>U</td>
<td>Undefined format.</td>
</tr>
<tr>
<td>V</td>
<td>Variable, unblocked.</td>
</tr>
<tr>
<td>VA</td>
<td>Variable, unblocked, ASA characters.</td>
</tr>
<tr>
<td>VBA</td>
<td>Variable, Blocked, ASA characters.</td>
</tr>
<tr>
<td>VBS</td>
<td>Variable, Blocked, Spanned.</td>
</tr>
</tbody>
</table>

VSE SAM does not automatically record a file's geometry (RECFM, LRECL and BLKSIZE) in disk VTOCs. However, it is common for VSE applications that perform I/O to include this information in the DTF control block and so write the file's geometry to its Format 1 record in the VTOC. CBLVCAT reports the record format, if present, otherwise blanks are displayed in the RECFM column.

---

**RECINP (nn)**

Display the number of records read since the file was defined or re-loaded.

The column width is 6, the format is '+nnnnK' and the heading is **RECS INPUT** (**RECS INPUT** also forms part of the **RECSTATS** combined column).

**Notes**

The file statistics are not always reliable, particularly if the file has been updated using control interval access.
Display the number of records inserted since the file was defined or re-loaded.

The column width is 6, the format is ‘+nnnnK’ and the heading is RECS INSRTD (RECS INSRTD also forms part of the RECSTATS combined column).

Notes
The file statistics are not always reliable, particularly if the file has been updated using control interval access.

Sets the number of records to be used for a tuning run.

It is useful when modelling new files, or when the catalog information for the file is believed not to be reliable (e.g. as a result of Control Interval access instead of logical record).

See GROWTH for a discussion of the interaction of the GROWTH and RECORDS parameters.
See Guide to VSAM Tuning for further details.

Display the record statistics for a file (DEL, INS, UPD and INP).

This field comprises four columns, each with the format ‘+nnnnK’: The column width is 24 and the overall heading is -- RECORD STATISTICS --.

Notes
The four columns which comprise this field are available separately as RECDEL, RECINS, RECUPD and RECINP.

The file statistics are not always reliable, particularly if the file has been updated using Control Interval access.

Display the number of records updated since the file was defined or re-loaded.

The column width is 6, the format is ‘+nnnnK’ and the heading is RECS UPDATD (RECS UPDATD also forms part of the RECSTATS combined column).

Notes
The file statistics are not always reliable, particularly if the file has been updated using control interval access.

For MVS systems only, REF may be used (instead of DDNAME) to select the required Catalog.
The argument of the REF parameter may be the catalog itself (self-defining entry), its ALIAS or the Data Set Name of any file in the Catalog (See also REF=ALL).

This method does not need a DD statement as it dynamically allocates the Catalog.

**REF=ALL**

(LISTVCAT)

<table>
<thead>
<tr>
<th>ALL</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC  REF=ALL</td>
</tr>
</tbody>
</table>

For MVS systems only, REF=ALL specifies that all catalogs are to be processed.

**Notes**

This feature is only available to ICF Catalogs with DFP 3.1 or higher.

Each Catalog is a separate report (For a customised report REF=ALL MERGE allows the user to produce a single report across the whole system. See example 16 and example 17 in Guide to List Output).

Users should note that REF=ALL has obvious storage and run-time considerations and should only be used for system wide files. For files across a small number of catalogs it is more efficient to use MERGE to combine several different LISTVCAT commands.

If the synonym ALL is used, it must follow immediately after the LISTVCAT command.

**REPORT**

(COMMAND)

| REPORT VCAT DSN 22 SORT NRECS TYPE RECINS RECDEL |
| REPORT VTOC DSN TYPE BLKSIZE LRECL |

REPORT VCAT/VTOC is used to specify the fields to appear in a customised report (Obviously the Catalogs/VTOCs have to be selected by subsequent LISTVCAT / LISTVTOC statements).

The order of the parameters in the REPORT list is the order in which they are displayed.

**Notes**

The REPORT fields remain in force for the rest of the run, or until reset by a subsequent REPORT statement.

If no parameters follow REPORT VCAT/VTOC the layout defaults to the standard report.

The maximum report width is 250 characters (Report width is not the same as heading width - see PAGEWIDTH). If the parameters specified combine to give a report width of greater than 250, Error 12 will occur. It should be noted that when CBLVCAT is calculating the report width from the report parameters it automatically adds a blank separator, it is this value which must not exceed 250.

Each REPORT parameter may be specified with a numeric value which will override the implicit column width. In this case no blank column separator is added as CBLVCAT assumes that this has been included in the numeric value (Therefore supplying the default values as numeric arguments will not produce the same output as if no argument is specified).

Specifying a column width greater than default will result in blanks being added to the right of the column (A useful technique to increase the gaps between columns).

Specifying a column width less than default will truncate the information for all columns except for DSN (No check is made for numeric or alpha-numeric fields and all truncation is made from the right). The full Data Set Name is printed and, if the length exceeds the column width, all other details follow on the next line (Unless SORT / SORTD is included as a parameter, in which case the Data Set Name will be truncated).

Specifying a number directly following REPORT VCAT/VTOC will set a left margin.

**SORT**

REPORT output can be sorted into ascending or descending order by specifying SORT or SORTD immediately prior to the field on which sorting is to commence. The quantity of output can be curtailed with the STOPAFT parameter.
MERGE

Allows the user to create a single customised report containing data from multiple Catalogs and/or VTOCs and/or VVDSs.

If no SORT or SORTD parameter is supplied, MERGE will sort on the first REPORT parameter.

See Guide to List Output for examples of Customised reports and Summary of Syntax for a full list of parameters, column headings and default widths.

---

**RKP (nn)**

```
REPORT VCAT   DSN LMAX   KL   RKP CISIZE
```

Display the Relative Key Position for KSDS and AIX files.

The column width is 4, the format is ‘+nnn’ and the column heading is RKP (RKP is also available as part of the combined column KL,RKP/BLK/IMB which forms part of the standard report).

**Notes**

The Relative Key Position is the displacement of the key in the record relative to position zero (i.e. RKP 10 is the eleventh byte of the record).

To display the Key length use KL.

---

**S/C (nn)**

```
REPORT VCAT   DSN SORTD   S/C
```

For VSE systems only, display the local share option and the primary space class.

The column width is 5, the format is ‘+nnn’ and the the heading is S/C.

If CBLVCSW9 X'01' bit is on and option S/C is in effect, then the column, S/C, forms part of the standard report. CBLVCSW9 X'01' bit is set off by default in order to display 4-byte years in report date fields (TIMESTAMP and DEFINED).

See mutually exclusive option, SHR and and description of option S/C.

---

**SECALLOC=nn**

```
LC    DD=CBLV01    SUBSET  SECALLOC=50     * If at least 50 sec extents
```

Select files whose number of secondary extents is greater than or equal to the value specified.

**Notes**

Specifying SECALLOC=0, however, will select files which have no secondary extents.

In a future release, SECALLOC will be replaced by a more meaningful parameter.
### SEV=n

**(LISTVCAT SUBSET)**

```
LC DD=CBLI05 KEY=/TEMP SEV=2
```

Select files depending upon CBLVCAT’s allocated SEVerity message level (i.e. according to their current state of tune).

**Notes**

The **SEV=n** subset does not cause tuning recommendations to be produced (See the SEVerity Block section in Tuning Output).

For a file selected at a particular severity level, all SEV messages relating to that file are printed.

### SEVL (nn)

**(REPORT VCAT))**

```
REPORT VCAT DSN SORTD SEVL
```

Display a file’s highest severity level and the lowest severity number at that level.

The column width is 4, the format is ‘**n-nn**’ and the column heading is **SEVL**.

**Note**

The **SEVL** column will be left blank if the file has no associated severity messages.

### SHR

**(LISTVCAT OPTION)**

```
LC DD=CBLV01 OPTION SHR * Override CBLVCSW1=X'08'
```

If **S/C** is the current installation default, **SHR** can be used to display the column containing the local (cross region) and cross system share options, instead of the column containing the local share option and primary space class (S/C). **S/C** and **SHR** are mutually exclusive options.

In the CBL supplied version of **CBLNAME**, **SHR** is the default (**CBLVCSW1** bit X’08’ is on). **S/C** can be made the default by changing this setting.

**Notes**

If **Share Option 4** is being used, CBLVCAT flags this with an asterisk, as it should be avoided due to its system overheads.

**VSE** operating systems prior to **VSE/ESA 1.3** does not support the cross system share option.

### SHR (nn)

**(REPORT VCAT)**

```
REPORT VCAT DSN SORTD SHR
```

Display the local (cross region) and cross system share options.

The column width is 5, the format is ‘**n,n**’ and the heading is **SHR**.

If **CBLVCSW9 X’01’** bit is on and option **SHR** is in effect, then the column, **SHR**, forms part of the standard report. **CBLVCSW9 X’01’** bit is set off by default in order to display 4-byte years in report date fields (TIMESTAMP and DEFINED).

See mutually exclusive **VSE** system option, **S/C** and and description of option **SHR**.

### SMS (nn)

**(REPORT VCAT)**

```
REPORT VCAT DSN SMS
```

Display the System Managed Storage (SMS) **Storage Class**, **Data Class** and **Management Class** attributes for the file.

The column width is 26, the format is ‘**xxxxxxxx xxxxxxx xxxxxxx**’ and the column heading is **SMSS SMSD SMSM**.
SMSD (nn) (REPORT VCAT)

Display the System Managed Storage (SMS) **Data Class** attribute for the file.
The column width is 8, the format is 'xxxxxxxx' and the column heading is **SMSD**.

SMSM (nn) (REPORT VCAT)

Display the System Managed Storage (SMS) **Management Class** attribute for the file.
The column width is 8, the format is 'xxxxxxxx' and the column heading is **SMSM**.

SMSS (nn) (REPORT VCAT)

Display the System Managed Storage (SMS) **Storage Class** attribute for the file.
The column width is 8, the format is 'xxxxxxxx' and the column heading is **SMSS**.

SORT=xxx (LISTVTOC OPTION)

<table>
<thead>
<tr>
<th>LV</th>
<th>VOL=UCATWK1</th>
<th>SORT=EXT</th>
<th>* By absolute extent</th>
</tr>
</thead>
<tbody>
<tr>
<td>LV</td>
<td>DEV=341</td>
<td>SORT=DSN</td>
<td>* By data set name</td>
</tr>
</tbody>
</table>

Used to select the order that files are to be displayed in the VTOC report.
The following arguments are supported:

**DATE**
Files will be printed in **descending** CREATE order (The filename is used as a secondary sort field - ascending order).

**DSN**
Files will be printed in **ascending** filename order.

**EXP**
Files will be printed in **ascending** EXPIRY date order. (The filename is used as a secondary sort field and the EXPD/NOEXPD does not have any effect on the order).

**EXT**
Files are sorted by absolute extent position for each element of the file (i.e. on each EXTENT of that file ). Additional extents will not necessarily follow immediately after the Prime extent, but will appear where they are located (which could even be in front of the Prime extent). EXT is the **default** sort order.

Secondary extents for multi-extent files will show the filename, the extent information, and also EXT=n in the **TYPE** column.

The **Prime** extent will have the literal 'MULTI-EXTENT' in the **INFO** column and the correct file type in the **TYPE** column.

**EXTPRIME**
Files will be printed after sorting by primary extent (logical file order). Additional extents will be listed under the prime extent.

**SIZE**
Files will be printed in **descending** order of file size.

Free space is now also sorted (if requested via **FREE** option).

**FREETAB** reporting will also be sorted in descending **SORT** order.
SORT  
(REPORT VCAT,VTOC)

SORTA

REPORT VCAT  SORT  DSN  TYPE  NRECS  TIMESTMP

Used to sort lines of Customised report output into ascending order.

Note

One specification of SORT is allowed per REPORT command. Its position within the list of keywords (defining the columns of the report) indicates where the sort begins. The sort is actioned on all the fields following the parameter, therefore, a sort of field within field can be achieved by placing the desired items contiguously in the parameter list (See also the description of STOPAFT).

The REPORT fields used (including SORT) are the ones applicable to the last LISTVTOC/ LISTVCAT command (See example 14 in Guide to List Output).

Where SORT/SORTD is not specified, the LISTVTOC or LISTVCAT report is produced in the order that entries are found in the VTOC or catalog respectively.

SORTD  
(REPORT VCAT,VTOC)

REPORT VTOC  DEV=ALL  SORTD  BLKSIZE  EXPIRES

Used to sort lines of Customised report output into descending order.

See Notes in REPORT parameter SORT.

SPANned  
(LISTVCAT SUBSET)

LC  DD=CBLV01  SUBSET  SPANNED

Select files defined with the SPANNED attribute.

SPLIT=nn  
(LISTVCAT SUBSET)

LC  DD=CBLV22  SUBSET  SPLIT=20

For KSDS and AIX files only, select files which have experienced at least 1 CA split or nn CI splits (see SPLITS in Additional VSAM Information for a discussion on CI and CA splits).

Notes

For the standard report the number of splits are reported in the TIMESTMP / DEFINED column if they exceed the warning values (CBLVCSCA and (CBLVCSCI).

For a Customised report the number of splits can be reported if either the SPLITCI, SPLITCA or SPLITS columns are displayed.

SPLITCA (nn)  
(REPORT VCAT)

REPORT VCAT  DSN 22  TYPE  SORTD  SPLITCA  SPLITCI

For KSDS and AIX files only, display the number of Control Area splits that have taken place since the file was defined (or re-loaded).

The column width is 5, the format is ‘+nnnn’ and the heading is SPLIT CA (SPLIT CA also forms part of the combined SPLITS column). See Notes under SPLITs.
SPLITCI (nn) (REPORT VCAT)

REPORT VCAT  DSN 22  TYPE SORTD SPLITCA  SPLITCI

For KSDS and AIX files only, display the number of Control Interval splits that have taken place since the file was defined (or re-loaded).

The column width is 5, the format is ‘+nnnn’ and the heading is SPLIT CI (SPLIT CI also forms part of the combined SPLITS column). See Notes under SPLIT.

SPLITS (nn) (REPORT VCAT)

REPORT VCAT  SORT DSN 22  TYPE ALLOC4  SPLITS

Display the number of both CI and CA splits that have taken place since the file was defined (or re-loaded).

The column width is 10, the format is ‘+nnnn+nnnn’ and the heading is SPLITS CI CA.

Notes

The columns are also available separately via the SPLITCA and the SPLITCI columns.

A descending SORT on this field will not necessarily result in the most fragmented files appearing at the top of the report, as the primary sort is on CI splits (To sort on CA splits, specify CA and CI as separate fields).

Standard reports display the number of CI and CA splits under the TIMESTMP / DEFINED column when the appropriate split threshold has been reached (CBLVCSCA or (CBLVCSCI).

See SPLITS in Additional VSAM Information for a discussion on CI and CA splits).

START (nn) (REPORT VTOC)

REPORT VTOC  DSN 22  SORT START  ALLOC  USED

Display the relative start address of the file.

The column width is 8, the format is ‘+nnnnnnn’ (tracks for CKD blocks for FBA) and the heading is START (START also forms part of the standard report).

STOPAFT=nnn (REPORT VCAT,VTOC)

REPORT VTOC  DSN 22  SORT ALLOC LRECL
LV VOL=CBLT03 STOPAFT=30
REPORT VCAT  DSN AVRL ALLOC3 NRECS
LC REF=ALL MERGE STOPAFT=40

For use with the REPORT VCAT/VTOCSORT parameter. It allows the user to limit the number of files to be included in the report.

Notes

STOPAFT=nnn, in conjunction with SORT and MERGE, is a useful method for reporting on the top nnn files of the sorted output across multiple Catalogs/VTOCs (The order of the output depends upon the position of the SORT parameter). However, use of MERGE in this context would diminish the information to nnn entries in total, rather than nnn for each catalog or VTOC (this of course could well be what is required). See example 16 in Guide to List Output.

If no SORT or SORTD parmater is supplied, MERGE will sort on the first REPORT parameter. Therefore, it is not strictly necessary to code SORT/SORTD in the REPORT statement.

STOPAFT only limits the number of files at print time. All the requested information will still be collected and sorted.
### STRIPED

**(LISTVCAT SUBSET)**

**STR**

```
LISTVCAT REF=CBL.EXT.X019991 STRIPED
```

For MVS, selects all data sets that have been defined with SMS STORAGECLASS having a non-zero Sustained Data Rate (SDR) value.

For VSE, subset parameter STRIPED is ignored.

### SUBSET

**(LISTVCAT,LISTVTOC)**

```
LC DD=CBLV01 SUBSET TYPE=K NRECS=2000
```

The keyword **SUBSET** indicates that report selection sub-parameters will follow.

**SUBSET** sub-parameters on a **LISTVTOC/LISTVCAT** operation allow you to restrict selection to a particular file (set of files).

**Notes**

The word **SUBSET** is optional as all its associated parameters are unique.

**SUBSET** parameters are processed as a logical **AND** (i.e. cumulative). The exceptions are **KEY** and **IGN**, which are processed as a logical **OR**.

Each **SUBSET** sub-parameter is only effective for the current **LISTVTOC/LISTVCAT** operation.

See **Summary of Syntax** for a list of all **SUBSET** parameters.

### SUMMARY

**(LISTVCAT)**

```
LC DD=CAT006 SUMMARY
```

Used to suppress the detail report and generate only the Catalog summary and the self-defining Catalog entry.

**Notes**

If the **SUMMARY** parameter is not specified, the Catalog summary is included automatically after the main detail report (unless the selection criteria result in no files reported). **SUMMARY** cannot be used in conjunction with **TUNE** or **SUBSET**.

The columns of the summary comprise as follows:

**VOLUME**

The volume serial number (The catalog owns the space on this volume which the rest of the line describes).

**CRA**

For **VSAM Catalogs only**, the catalog recovery area (For non-recoverable catalogs this will contain ‘NO’ and for recoverable catalogs, the CI number of the recovery area).

**TYPE**

The disk device type of the associated volume (MVS virtual disks on Mass Storage Systems will be shown with a V suffix e.g. 3380V).

**MIN-CA**

The range of MiNimum Control Area sizes (bytes for CKD, blocks for FBA).

For FBA devices there is only one value displayed, whereas for CKD devices, the range shows the upper and lower limits that may be achieved depending upon the physical record size chosen by VSAM (but not necessarily in the level of the Operating System on which CBLVCAT is running).

**MAX-CA**

The range of MAXimum control area sizes (bytes for CKD, blocks for FBA).

For FBA devices there is only one value displayed, whereas for CKD devices, the range shows the upper and lower limits that may be achieved depending upon the physical record size chosen by VSAM (but not necessarily in the level of the Operating System on which CBLVCAT is running).

**DATA SETS**

A count of the number of data sets belonging to this catalog on the volume (excluding the catalog itself).
Alternate Index (AIX) files, and Key Sequence Data Set (KSDS) files are comprised of an index component and a data component, both are included in this value.

When a file is defined on multiple volumes, each file occurrence will be included in that volume’s total.

Files which are defined and have space allocated, but are empty, are also counted.

**SPACE CLASS**
For **VSE** VSAM catalogs only, shows the space class (Each different space class residing on a volume has a line of detail printed from this field onwards).

**DATA SPACES**
The number of VSAM data space areas on this volume (This item does not apply to ICF catalogs). For **VSE** each different space class has a separate entry.

**TRACKS or BLOCKS**
There are five sub-columns under this group column heading:

<table>
<thead>
<tr>
<th>ALLOC</th>
<th>The number of tracks (CKD) or blocks (FBA) allocated to this space on this volume.</th>
</tr>
</thead>
<tbody>
<tr>
<td>USED</td>
<td>The total number of tracks/blocks allocated to data sets in this space on this volume (This value should match that of the 'TOTAL ALLOC' column total of the detail report).</td>
</tr>
<tr>
<td>PCNT</td>
<td>The total number of tracks/blocks allocated to data sets in this space on this volume. The percentage of total space available occupied by data sets (It represents the number of records in the file compared with the capacity of the current total allocation). For <strong>KSDS</strong> files, the space free is fully usable only if all the future inserts have keys that are evenly spread throughout the file. When the value under this column has two asterisks on either side (see example 3 in Guide to List Output), it is an indication that the VOLUME PERCENT WARNING threshold has been reached (This threshold value, 85% by default, may be changed using <strong>CBLVCPCV</strong> either at run-time or in <strong>CBLNAME</strong>).</td>
</tr>
<tr>
<td>FREE</td>
<td>The number of tracks (or blocks) remaining in the data space (space available to expand existing files or allocate new ones). <strong>VSE</strong> users can also use this value to see if enough space remains for temporary work files (see <strong>NOALLOC</strong> below).</td>
</tr>
<tr>
<td>MAXF</td>
<td>Displays the largest contiguous free extent available per ‘space class’. This field is intended to assist with planning the location of medium to large files. It will not always be exact and in some cases can be up to 14 tracks too small.</td>
</tr>
</tbody>
</table>

**TRACKS USED**
The number of tracks on the volume taken up by files controlled by the catalog (ICF only). This is the only allocation information in an ICF summary.

**TIMESTAMP**
The date when a VSAM data space was last added to or deleted from the volume (format is **ccyy/mm/dd hh.mm.ss**). This value does not apply to ICF catalogs.

**NOALLOC=nnn**
The date when a VSAM data space was last added to or deleted from the For **VSE** systems only. The total space which will be required by temporary WORK files. This non-standard report line appears beneath and to the right of the **FREE** sub-column item.

Work files are only allocated when they are required, and the space they need is taken from the unused pool.

It is unusual for all temporary files to be opened at the same time.

The **NOALLOC** value is the sum of the primary allocation for all **TEMP** files in the detail report.

---

**SUMMARY** *(LISTVTOC)*

**LV VOL=CBLT03 SUMMARY**

For the **Standard report only**, used to suppress the detail VTOC report and generate only the VTOC summary.
If the **SUMMARY** parameter is not specified, the VTOC summary is automatically included, after the main files report, in the **INFO** column (at the same time as the freespace if **FREETAB** is active). However, the **SUMMARY** is not printed for **SUBSET** runs.

If the **SUMMARY** parameter is used, the "**OLDEST**" and "**LATEST**" fields will be missing from the **SUMMARY**. Only the **SORT** and **DEV=ALL** sub-parameters may be used with **SUMMARY**.

---

**SYS=ALL**  
**(LISTVTOC)**

<table>
<thead>
<tr>
<th>LV</th>
<th>EJ=YES</th>
<th>* New page for each report</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYS=ALL</td>
<td>* All assigned drives</td>
<td></td>
</tr>
</tbody>
</table>

For **VSE** systems only, produces a VTOC report for every disk drive that has a current assignment in the range SYS000 - SYS254. To obtain VTOC listings of all active disk drives use **DEV=ALL**.

**Notes**

Devices which are not READY (online) are not included in the report.

---

**SYS=nnn**  
**(DEL,LISTVTOC,MOD)**

| DEL SYS=04 | * Choose device (requires ASSGN SYS004....) |
| DEV=152 | * Check assignment |
| DSN=ABC.PROD.FILE | * No quotes required |
| FAIL=EOJ | * If assignment wrong etc |

For **VSE** systems only, the **DEV** and/or **SYS** and/or **VOL** parameters determine the current volume. When a combination of two or more of these parameters is specified, the order of checks for conformance are as follows:

**DEV and SYS**

The assignment specified by **SYS** must match the cuu specified by **DEV** otherwise **ERROR 28** is incurred.

**DEV and VOL**

The volser specified by **VOL** must match the volume mounted on the cuu specified by **DEV** otherwise the **OPERATOR** is prompted to mount the correct volume.

**SYS and VOL**

The volser specified by **VOL** must match the volume mounted on the DASD assigned to by **SYS** otherwise the **OPERATOR** is prompted to mount the correct volume.

If a check fails then the operation will fail, with subsequent processing depending on the **FAIL** parameter.

**Notes**

If **DEV**, **SYS** and **VOL** are omitted, the drive will be that last used or will default to **SYS=0** if no previous device specified.

Within a single invocation of **CBLVCAT**, a second or subsequent **LISTVTOC** operation on the same volume(s) will benefit from significant run-time improvements, as it uses previously stored information.

See also the **OWN** parameter for checking that the correct volume is mounted.

---

**TIMESTAMP**  
**(LISTVCAT OPTION)**

**TIMESTAMP**

| LC | DD=CBLV01 | OPTION | TIMESTAMP | SUBSET | LODATE=93 |

If **DEFINED** is the current installation default, **TIMESTAMP** can be used to display the date the file was last closed by an operation that may have changed its contents, instead of the definition date. (**TIMESTAMP** and **DEFINED** are mutually exclusive options).

In the **CBL** supplied version of **CBLNAME**, **TIMESTAMP** is the default (**CBLVCSW1** bit X’02’ is on). **DEFINED** can be made default by changing this switch.
Example: LISTVCAT TIMESTMP field (CBL Ref: vmxtime)

Example: LISTVCAT TIMESTMP field (CBL Ref: vmxtime)

Note
This OPTION also governs the date that will be used in a HIDATE, LODATE SUBSET.

### TIMESTMP (nn) (REPORT VCAT)

The column width is 20 and the heading is TIMESTMP (TIMESTMP also forms part of the standard report if the option TIMESTMP / DEFINED is set to TIMESTMP).

Where included as part of the LISTVCAT standard report, TIMESTMP has the format ‘*ccyy/mm/dd hh.mm.ss’. However, if specified on REPORT as part of a customised report, then the format is ‘ccyy/mm/dd hh.mm.ss’ with ‘*’ (asterisk) following the date as opposed to preceding it. This is so that SORT on the TIMESTMP field is reliable.

Notes
If the file has been defined but not loaded, there is no timestamp. In this case CBLVCAT reports the DEFINED date (This is in ccyy/mm/dd format and is therefore easily recognisable). However, if a file is defined, loaded with one record, and then that record is deleted (making the file empty), the full timestamp is shown.

The timestamp is held as an absolute value to GMT (Greenwich Mean Time). Users on other time zones may set CBLVCSW9=X’02’ (CBLNAME option VTZAdjust=Yes) to adjust the time and date to the local time zone.

If CBLVCSW9=X’01’ is set on (CBLNAME option V2digitYear=Yes) then the format of the TIMESTMP field is ‘yy/mm/dd hh.mm’ or ‘yy/mm/dd hh.mm*’ for standard or customised reports respectively. Note that this option will also include report field SHR or S/C as part of the standard report.

**ccyy/mm/dd hh.mm.ss**
An asterisk eye-catcher preceding the time stamp value (or following it in a customised report) indicates that the KSDS index is out of step with its data component (possibly caused by file corruption, or a program processing and updating the index or data components separately).

The timestamp value may not be accurate if the file being reported is currently in use by another program, as it is not updated until file closure.

‘***nnn CI SPLITS***’ and ‘***nnn CA SPLITS***’
Non-standard lines appearing in the Standard report if the appropriate splits threshold has been reached (CBLVCSCA or CBLVCSCI).

### TOTALLOC=nnn (LISTVCAT SUBSET)

| TOTALLOC=n.nn | LC DD=CBLV01 SUBSET TOTALLOC=0 | SECALLOC=10 * If at least 400 trks and 10 sec extents. | LC DD=CBLV01 SUBSET TOTALLOC=00 | * List temp files |

Select files with a total allocation greater than or equal to the value specified (tracks for CKD, blocks for FBA).

Note
VSE users may specify TOTALLOC=0, in order to obtain a report of dynamic files defined with the IDCAMS attributes NOALLOCATION and REUSE (They are normally used as compiler work files). These files appear on the report with "TEMP" in the TOTAL column. Space is allocated when they are initially opened for output and deleted once they are closed (depending on the ACB or JCL options).
TOTALS

<table>
<thead>
<tr>
<th>LC</th>
<th>DD=CBLI04</th>
<th>KEY=ABC</th>
<th>KEY=ABE</th>
<th>TOTALS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC</td>
<td>DD=CBLV03</td>
<td>NRECS=0</td>
<td>TOTALS</td>
<td>* Empty files</td>
</tr>
</tbody>
</table>

Display only the totals for the datasets selected by suppressing the reporting of individual datasets.

```
SPACE ALLOCATED TO TEST FILES ON CBLV03 AND CBLV04  2009/08/19  PAGE   1
-------------------------------------------------------------------------------------
<table>
<thead>
<tr>
<th>REPORT</th>
<th>VCAT</th>
<th>ALLOCT</th>
<th>ALLOCP</th>
<th>ALLOCS</th>
<th>CISIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISTVCAT DD=CBLV03  KEY=TEST  TOTALS  MERGE  HEAD='SPACE ALLOCATED TO TEST FILES ON CBLV03 AND CBLV04'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LISTVCAT DD=CBLV04  KEY=TEST  TOTALS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ALLOC  ALLOC  ALLOC  CISIZE
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>130244</td>
<td>73619</td>
<td>2638</td>
<td></td>
</tr>
</tbody>
</table>

CBLVCAT IS LICENSED BY COMPUTE (BRIDGEN) LTD 0656-652222, 0656-656466
** EXPIRY DATE --- 6 JUN 1995 **
```

Example: TOTALS (CBL Ref: vbxv5f3)

Notes

TOTALS can be applied to Standard or Customised Reports. The columns to which it applies are:
ALLOC, ALLOC, ALLOCS, SPLITCA, SPLITCI and ALLOC.

All other specified report items (e.g. CISIZE) are left blank, however, the column headings are still printed.

When used with MERGE, the user is able to obtain totals across multiple catalogs without reporting on the individual catalog detail (The HEAD='string' parameter enables the user to give this type of report a descriptive heading).

TUNE (sys)

<table>
<thead>
<tr>
<th>LC</th>
<th>DD=CBLV01</th>
<th>KEY=PROD.WRK1</th>
<th>TUNE</th>
<th>DEFINE</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC</td>
<td>DD=CBLV22</td>
<td>TUNE=MVS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Select files with a total allocation greater than or equal to the value specified (tracks for CKD, blocks for FBA).

The TUNE parameter will cause tuning recommendations to be added to the report and, if DEFINE is specified, the IDCAMS DEFINE output.

TUNE without a sys argument tunes for the current operating system. If a sys argument is used, tuning recommendations are based upon that argument.

Notes

TUNE is only active for the operation on which it is specified.

For CMS users, if no sys argument is used, tuning recommendations will be for the MVS operating system (unless DOS is set ON when they will be for VSE). The title page of the Detail Report identifies the system in effect.

Allowable sys arguments are MVS and VSE.

When no TUNE sub-parameters are specified, unrestricted tuning recommendations will be given. However, if run-time overrides are supplied (e.g. RECORDS=nnn), the tuning recommendations will reflect these overrides (unless they are unreasonable).

Files that do not require tuning are shown as 'FILE ALREADY IN TUNE'. A file that is already in tune might have SEV 1-nn messages indicating fine tuning conditions.

See Guide to VSAM Tuning for further details and examples.
Display the file type (e.g. DA, PDS, VSAM etc).

The column width is 7, the format is ‘xxxx’ or ‘*EXT=n’ and the column heading is TYPE (TYPE also forms part of the standard report).

Possible entries in this column are as follows:

<table>
<thead>
<tr>
<th>(blanks)</th>
<th>The field is left blank for sequential (SD) files that do not occupy split extents.</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>A sequential file (PS) has been defined as <strong>unmovable</strong> (In VSE systems the file has been created using a DLBL of IJSYSxx).</td>
</tr>
<tr>
<td>nnnn</td>
<td>For CKD disks only, indicates the volume device type, e.g., 3380, 3390.</td>
</tr>
<tr>
<td>DA</td>
<td>The file was created using the Direct Access method.</td>
</tr>
<tr>
<td>DAU</td>
<td>A direct access file (DA) has been defined as <strong>unmovable</strong>.</td>
</tr>
<tr>
<td>EXT=n</td>
<td>The extent number of a multi-extent file.</td>
</tr>
<tr>
<td><em>EXT=n</em></td>
<td>The extent with the lower extent number has not been processed (possibly on another volume). This will not be given when the report is listed by absolute extent (i.e. <strong>SORT=EXT</strong>).</td>
</tr>
<tr>
<td>FBA</td>
<td>Indicates that the device is of Fixed Block Architecture.</td>
</tr>
<tr>
<td>PDS</td>
<td>The file is a Partitioned Data Set.</td>
</tr>
<tr>
<td>PDSE</td>
<td>The file is a Partitioned Data Set Extended.</td>
</tr>
<tr>
<td>PDU</td>
<td>The Partitioned Dataset (PD) has been defined as <strong>unmovable</strong>.</td>
</tr>
<tr>
<td>SPLIT</td>
<td>A sequential file occupying a Split cylinder.</td>
</tr>
<tr>
<td>VSAM</td>
<td>The file is a VSAM data space.</td>
</tr>
</tbody>
</table>

Display the file type (e.g. KSDS, NONVSAM, SAM etc).

The column width is 8 and the column heading is TYPE (TYPE also forms part of the standard report).

Possible entries in this column are as follows:

<table>
<thead>
<tr>
<th>(blanks)</th>
<th>The field is left blank for sequential (SD) files that do not occupy split extents.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-Z</td>
<td>Reference Notes</td>
</tr>
<tr>
<td>2013-10-29 15:43:01 CBLVCAT Manual 125</td>
<td></td>
</tr>
</tbody>
</table>
### AIX
An Alternate Index (This will be followed by an index (IX) line and possible association details). The key position applies to its location in the associated KSDS.

### (G)
An Alternate Index Dataset defined with the **UPGRADE** attribute.

### (Q)
An Alternate Index Dataset defined with the **UNIQUEKEY** attribute.

### (QG)
An Alternate Index Dataset defined with the **UPGRADE** and **UNIQUEKEY** attributes.

### ALIAS
For a Customised report the alias name will be displayed. However, for a Standard report the name of the referenced file is also shown (the alias will also be shown with other aliases against the NONVSAM, USERCAT or ICFCAT file entry - see **ALIAS** for an example). Alias detail lines can be suppressed (see **NOALIAS**).

### ESDS
An Entry Sequenced Data Set.

### GDG
A Generation Data Group. For the standard report only, the base name, the maximum number of generations, the version and generation numbers of associated NONVSAM files together with disk and device type, or tape sequence number, are also shown (see below). Access to this information in a Customised report is via the **VOLn** fields.

### ICFCAT
An ICF connector record. For a Standard report the volume and aliases of ICF catalogs linked to this catalog are also shown.

### IX
The Index component of an AIX, KSDS or VRDS file. For a VVDS report the data structure is also displayed. (i.e. KSDS, VRDS or AIX).

### KSDS
The Data component of a Key Sequenced Data Set (This line will be followed by an index (IX) line which describes the INDEX component and, possibly, association detail lines).

### (R)
A reusable file (When it is opened for output it will effectively become a new file).

### LDS
A LINEAR DATA SET.

### NONVSAM
MVS files that do not occupy space within the catalog. For a standard report volume details (if not on the catalog volume) and any associated aliases are also shown. Details are not printed here for files that form part of a GDG.

### PAGESP
OS system page space (similar to ESDS lines).

### RRDS
A Relative Record Data Set.

### SAM
For **VSE** only, a Sequential Access Method file managed by VSAM (This is similar to ESDS except that **BLKSIZE** contains the blocksize of the SAM file).

### USERCAT
A User catalog. For the standard report the volume and aliases of VSAM catalogs linked to this master catalog are also shown. (When the User Catalog resides on an emulated disk, the device type is suffixed with a V, e.g. USERCAT **VOL1=CBLV0X 9335V**).

### VRDS
The Data component of a Variable-length Relative Record Data Set. This line will be followed by an index (IX) line which describes the INDEX component.

---

### TYPE=xxx

#### (LISTVCAT SUBSET)

**TY**

| LC DD=UCAT1 |
| SUBSET TYPE=AC | * Select AIX and Clusters. |

| LC DD=UCAT2 |
| SUBSET TYPE=ACNUX | * Select all except GDG. |

Select files based on their type.

**Notes**

Any combination of the following types is allowed:
<table>
<thead>
<tr>
<th>A</th>
<th>Alternate Index file</th>
<th>AIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Cluster (the same as EKLPRSV)</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>Entry Sequenced Data Set</td>
<td>ESDS</td>
</tr>
<tr>
<td>G</td>
<td>Generation Data Group</td>
<td>GDG (MVS only)</td>
</tr>
<tr>
<td>K</td>
<td>Key Sequenced Data Set</td>
<td>KSDS</td>
</tr>
<tr>
<td>L</td>
<td>Linear Data Set</td>
<td>LDS (MVS only)</td>
</tr>
<tr>
<td>M</td>
<td>VSAM (the same as AEKRV)</td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>Non-VSAM file.</td>
<td>NONVSAM (MVS only)</td>
</tr>
<tr>
<td>P</td>
<td>PAGE Data Space</td>
<td>PAGESP (MVS only)</td>
</tr>
<tr>
<td>R</td>
<td>Relative Record Data Set</td>
<td>RRDS</td>
</tr>
<tr>
<td>S</td>
<td>SAM files managed by VSAM</td>
<td>SAM (VSE only)</td>
</tr>
<tr>
<td>U</td>
<td>User catalog</td>
<td>USERCAT/ICFCAT</td>
</tr>
<tr>
<td>V</td>
<td>Variable Length Relative Record Data Set</td>
<td>VRDS</td>
</tr>
<tr>
<td>X</td>
<td>Alias</td>
<td>ALIAS OF (MVS only)</td>
</tr>
</tbody>
</table>

**TYPE=xxx** *(LISTVTOC SUBSET)*

```
TY
    LV  VOL=CBLT01
    SUBSET  TYPE=DIPS  * Select all except VSAM
```

Select files based on their type.

**Notes**

Any combination of the following types is allowed:

<table>
<thead>
<tr>
<th>D</th>
<th>Direct Access files</th>
<th>DA</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>ISAM files</td>
<td>PRIME/INDEX/OFLOW</td>
</tr>
<tr>
<td>P</td>
<td>Partitioned Data Sets (PDS and PDSE)</td>
<td>PDS (MVS only)</td>
</tr>
<tr>
<td>P1</td>
<td>Partitioned Data Sets (PDS only)</td>
<td>PDS (MVS only)</td>
</tr>
<tr>
<td>P2</td>
<td>Partitioned Data Sets (PDSE only)</td>
<td>PDS (MVS only)</td>
</tr>
<tr>
<td>S</td>
<td>Sequential files</td>
<td>(left blank)</td>
</tr>
<tr>
<td>V</td>
<td>Virtual Storage Access Method</td>
<td>VSAM</td>
</tr>
</tbody>
</table>

**UNALLOC=nnn** *(LISTVCAT SUBSET)*

```
LC  DD=CBLV01
    SUBSET  UNALLOC=15  * If at least 15 blks/trks
```

Select files whose unused allocation is greater than or equal to the specified value (tracks CKD, blocks FBA).

**Notes**

Once a control area contains at least one record it is considered allocated, therefore UNALLOC=0 selects files which have at least one record in all of its control areas.

A comparison of unallocated space before and after a reload with no change of definition can show an increase in unallocated space (i.e. Distributed freespace can be recovered as unallocated space).

**UNIT (nn) - REPORT VTOC** *(REPORT VTOC)*

```
REPORT VTOC  VOLUME  SORT UNIT  DSN  CREATED  ACCESSED  EXPIRES
```

Display the device channel and unit (c uu) of the volume being processed.
Example: VTOC Report with REPORT field UNIT (CBL Ref: vvtunit)

The column width is 4, the format is 'ccuu' and the column heading is UNIT.

### UNUSED

**LISTVCAT OPTION**

If **PCNT** is the current installation default, **UNUSED** can be used to display the amount of unused space in the current allocation, instead of the percentage of space used in the current allocation (**PCNT**). **UNUSED** and **PCNT** are mutually exclusive options.

In the CBL supplied version of **CBLNAME**, **PCNT** is the default (**CBLVCSW1** bit X'04' is on). **UNUSED** can be made the default by changing this setting.

**Notes**

As this value relates to the current allocation, it is possible (for KSDS/AIX with distributed freespace) that a redefined and loaded file could show an increase in this value.

**UNUSED** can be shown on a Customised report as part of the **ALLOC4** field.

### USED (nn)

**REPORT VTOC**

Display the number of tracks/blocks actually used in the extent.

The column width is 7, the format is +nnnnnn (tracks for CKD or blocks for FBA) and the heading is **USED**. (USED also forms part of the standard report).

**Notes**

CBLVCAT calculates the value for the VTOC itself.

For **VSE** systems, this field is normally maintained for ISAM files only.
Non-ISAM files will only have an entry if the installation has software to update the 'last record' field in the format 1 label in the VTOC.

VCAT (nn)  (REPORT)

Valid on the REPORT operation only, it indicates that the subsequent parameters describe the columns of a customised Catalog Report.

Notes

Supplying "nn" sets the left margin at column 'nn', otherwise the margin defaults to column 0.

Resetting to the standard report may be achieved by supplying REPORT VCAT without additional parameters.

See also REPORT.

VOL=volser  (LISTVCAT SUBSET)

Select files which reference the specified volume.

Notes

If a file references more than one volume, all the other volume lines will be printed.

Specifying CBLVCET=n (n non-zero) will show the physical address of each extent used on the volume.

VOL=volser  (DEL,MOD)

For VSE systems only, used to select or verify the required volume.

Notes

The disk volume label may be modified via MOD NEWVOL=xxx. However, in this instance, VOL is used only as a check that the correct disk has been selected, i.e. a DEV and/or SYS parameter must also be coded.

WARNING - Take great care when modifying disk volume serial numbers.

See also DEL and MOD.

VOL=volser/vol*  (LISTVTOC)

Selects the required VTOC by direct reference to the volume serial number.

Notes

For MVS and VSE systems, The VOL=vol* syntax produces a separate VTOC report for every volume which has a generic name match (VOL=vol* in conjunction with MERGE would combine the separate reports into a single listing).

VOL=* is exactly the same as DEV=ALL

For MVS and VSE systems, Within a single invocation of CBLVCAT, a second or subsequent LISTVTOC operation on the same volume(s) will benefit from significant run-time improvements, as it uses previously stored information.

For VSE systems only, the DEV and/or SYS and/or VOL parameters determine the current volume. When a combination of two or more of these parameters is specified, the order of checks for conformance are as follows:
DEV and SYS
The assignment specified by SYS must match the cuu specified by DEV otherwise ERROR 28 is incurred.

DEV and VOL
The volser specified by VOL must match the volume mounted on the cuu specified by DEV otherwise the OPERATOR is prompted to mount the correct volume.

SYS and VOL
The volser specified by VOL must match the volume mounted on the DASD assigned to by SYS otherwise the OPERATOR is prompted to mount the correct volume.

If a check fails then the operation will fail, with subsequent processing depending on the FAIL parameter.

If DEV, SYS and VOL are omitted, the drive will be that last used or will default to SYS=0 if no previous device specified.

See also the OWN parameter for checking that the correct volume is mounted.

---

**VOLINFO** *(LISTVCAT OPTION)*

```plaintext
LC DD=CBLI03 NOINDEX VOLINFO
```

If NOVOL is the current installation default, VOLINFO can be used to display the volume information lines (VOLINFO and NOVOL are mutually exclusive options).

In the CBL supplied version of CBLNAME, VOLINFO is the default (CBLVCSW1 bit X'10' is on). NOVOL can be made default by changing this switch.

```plaintext
  LISTVCAT DD=CBLV04 KEY=TSTX NOASSOC VOLINFO
  USERCAT CBLV04 (3380) TYPE NRECS PCNT ------ ALLOC TRACKS ---- FRSP
  -------------- ------ ------ ----- ----- ----- ----- ----------
  TSTX.ASSOC.TRANS.XREF KSDS 1023780 61.5 C=150 C=120 C=10 30 10
    VOL1=CBLV04
    CANDIDATE VOL2=CBLV32
  IX 153 68.0 5 5 1
    VOL1=CBLV04
    CANDIDATE VOL2=CBLV32
  ---- ---- ---- ---          
  ---- ---- ---- ---          
  ** ** ** ** ** ** CBLVCAT 2.12.156 Licensed by Compute (Bridgend) ** Expiry: 2010-07-20 **
```

Example: LISTVCAT volume details (CBL Ref: vmxvoln)

**Note**

Specifying CBLVCEXIT=n will force VOLINFO

`'VOLn=volser'`, relates to the preceding file detail line. It appears when a file resides on more than one volume (or a volume other than the catalog volume).

`'CANDIDATE'`, the file is eligible to be expanded onto this volume or another volume, but has not grown sufficiently to do so.

`'OVERFLOW'`, the file is using a volume that was not originally specified in the define (for files defined with the KEYRANGE attribute).

---

**VOL1 (nn)** *(REPORT VCAT)*

```plaintext
REPORT VCAT SORTD TIMESTAMP VOL1 50
```

Display the volume serial number of non-VSAM files.

The column width is 6, the format is `nnnnnn` and the heading is VOLSER.
Example: REPORT VCAT VOL1 (CBL Ref: vmxvol1)

<table>
<thead>
<tr>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifying a larger field width than the default value (which is one serial number per line) will allow as many serial numbers as possible to appear on a line within the given field width. Any remaining are carried over onto subsequent lines.</td>
</tr>
</tbody>
</table>

VOL2 (nn) (REPORT VCAT)

Display the volume serial number and device type for non-VSAM files.

The column width is 15, the format is ‘nnnnn xxxxxxxx’ and the heading is VOLSER DEVICE.

Example: REPORT VCAT VOL2 (CBL Ref: vmxvol2)

<table>
<thead>
<tr>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifying a larger field width than the default value (which is one data item per line) will allow as many combinations as possible to appear on a line within the given field width. Any remaining are carried over onto subsequent lines.</td>
</tr>
<tr>
<td>If the device is a tape unit, ‘TAPE’ will be displayed for the device type.</td>
</tr>
<tr>
<td>If CBLVCSW9 bit X'08' is ON, the DEVICE column displays devices in their internal hexadecimal notation.</td>
</tr>
<tr>
<td>In the CBL supplied version of CBLNAME CBLVCSW9 bit X'08' is OFF.</td>
</tr>
</tbody>
</table>

VOL3 (nn) (REPORT VCAT)

Display the sequence number and volume serial number for non-VSAM files.

The column width is 10, the format is ‘nnn nnnnnn’ and the heading is SEQ VOLSER.
Notes

Specifying a larger field width than the default value (which is one data item per line) will allow as many combinations as possible to appear on a line within the given field width. Any remaining are carried over onto subsequent lines.

Notes

Specifying a larger field width than the default value (which is one data item per line) will allow as many combinations as possible to appear on a line within the given field width. Any remaining are carried over onto subsequent lines.
Example: REPORT VCAT VOL5 (CBL Ref: vmxvol5)

Notes

Specifying a larger field width than the default value (which is one data item per line) will allow as many combinations as possible to appear on a line within the given field width. Any remaining are carried over onto subsequent lines.

If the device is a tape unit, 'TAPE' will be displayed for the device type.

If CBLVCSW9 bit X'08' is ON, the DEVICE column displays devices in their internal hexadecimal notation. In the CBL supplied version of CBLNAME CBLVCSW9 bit X'08' is OFF.

### VOLUME (nn)

<table>
<thead>
<tr>
<th>REPORT VTOC</th>
<th>VOLUME 8</th>
<th>DSN 18</th>
<th>SORTD ALLOC LRECL RECFM</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC</td>
<td>DD=DBLI11</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Display the volume serial number of the catalog volume (for LISTVCAT DDNAME), the VVDS volume (for LISTVCAT VVDS), or the VTOC volume (for LISTVTOC).

The column width is 6, and the column heading is 'VOLUME'.

Notes

VOLUME is designed for multiple operation runs, where post processing is much easier if there is a volume reference on each file line.

### VTOC (nn)

<table>
<thead>
<tr>
<th>REPORT VTOC</th>
<th>DSN 18</th>
<th>SORTD ALLOC BLKSIZE LRECL RECFM</th>
</tr>
</thead>
</table>

Valid on the REPORT VTOC operation only, it indicates that the subsequent parameters describe the columns of a customised VTOC Report.

Notes

Supplying "nn" sets the left margin at column 'nn', otherwise the margin defaults to column 0.

Resetting to the standard report may be achieved by supplying REPORT VTOC without additional parameters.

See also REPORT.

### VVDS=volser

| LC | VVDS=CB9009 |
| LC | VVDS=CB9123 | KEY=TEST |

List the catalog entries for an ICF VVDS (VSAM Volume Data Set).
Example: LISTVCAT VVDS=volser (CBL Ref: vbvx5f8)

Notes

The syntax and JCL requirements are the same as for the DDNAME parameter (except for TUNE and DEFINE which are not valid for VVDS reporting).

Parameters that request data not contained in the VVDS are ignored without generating an error (e.g. although GDG files can be requested with a TYPE parameter, it has no entries in the VVDS and are therefore not displayed). SMS managed volumes however, do have limited non-VSAM entry information in the VVDS and so is included in the VVDS report.

VVDS reporting is intended for problem situations (e.g. if the BCS is corrupt it is still possible to obtain the dataset information contained in the VVDS). For normal catalog reporting, via the BCS, see the DDNAME and REF parameters.

VVDS reporting can also be useful as part of a housekeeping operation (see examples in Guide to VSAM Tuning).

The VVDS is itself a VSAM ESDS file, so entries are displayed by CBLVCAT in the order they are contained in the VVDS (unless a SORT parameter is used on a REPORT VCAT command).

For KSDS and AIX files, the DATA and INDEX components are treated as separate entities (In fact, they do not have to exist in the same VVDS), consequently CBLVCAT reports the DATA and INDEX components as separate items even when both are present in the same VVDS (see above). This is in contrast to catalog reporting which reports the cluster as a single item.

For INDEX components (IX), the data type (KSDS or AIX) is also displayed in the TYPE field.

XVSAM

(LISTVCAT SUBSET)

For MVS, selects only VSAM data sets that have been defined with SMS DATACLASS attribute DSNTYPE=EXT. (XVSAM is a synonym for EXTENDED TYPE=M).

For VSE, XVSAM is a synonym for EXT-ADDR.
CBLNAME

This chapter illustrates and explains the CBLNAME module, which is the loadable phase/module that must be available to CBLVCAT at every execution (it provides installation standards and defaults).

The separate "CBLVCAT Installation Guide" (which is supplied with the Distribution Material), gives full information on how to set up the CBLNAME phase/loadmodule.

Introduction

CBL products SELCOPY and CBLVCAT load the MODULE/PHASE CBLNAME at startup to establish environment options.

CBLNAME Source Code

A skeleton CBLNAME source file is included as part of the product bundle.

In earlier releases, the CBLNAME source was simply an Assembler CSECT containing DC (define constant) for each field name (e.g. CBLVCSW1) and options were set by manually coding an operand value appropriate to the particular field. e.g.

```
CBLVCSW2 DC X'1C' Set Options FREETAB (X'18') & NOEXPD (X'04')
```

This method has been simplified with the distribution of the CBLNAME Assembler Macro. The modern format of the CBLNAME source includes a single invocation of the CBLNAME macro with comma separated parameters representing each CBLNAME option field.

```
CBLNAME       Site='Compute (Bridgend) Ltd - Wales UK',               +
              VPassword='1234,5678,9ABC,DEF0',                        +
              VDateRange=(1975/07/11-2005/07/12),                     +
              VLineSpace=1,                                           +
...              
```

Throughout this manual, CBLNAME options are still referred to by their CSECT field names. However, the CBLNAME macro option format should be used to update and re-assemble CBLNAME.

CBLNAME Licensing

In early releases of CBLVCAT an expiry date was hard coded with in the CBLVCAT source code. This has been since been removed and replaced with a password mechanism which is based on the user's Company name/location and Operational Date Range(s) as supplied by CBL.

Note that the CBLNAME password is not based on CPUID.

The licence details are not dependent upon the release of the software and distributed independently of any new release of CBLVCAT. Therefore, it is possible (and encouraged) that the latest release of CBLVCAT be installed without having to apply new a password.

The user's Site string and new date range and password is distributed to all licensed users each year, before the end of the current operational date range, in order to allow continued successful execution of the software. These new licence details must replace existing licence details in the CBLNAME PHASE/MODULE.

In order to accommodate the new licence details, extensions have been added to the CBLNAME module for each CBL licensed product. Each extension has a variable length and contains fields for the products' operational date range(s) and user specific password.

These extensions are not easily inserted manually and so the CBLNAME macro should be used to avoid confusion and errors.

Please refer to the "CBL Products Install" documentation for comprehensive instructions on updating and maintaining CBLNAME.

Field Summary

The following table contains a summary of CBLNAME Macro parameters and arguments that apply to CBLVCAT, and their equivalent CSECT field names and offsets. Where applicable, default arguments are highlighted or enclosed in parentheses following.

<table>
<thead>
<tr>
<th>CBLNAME Macro Parameter</th>
<th>CBLNAME CSECT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site='string'</td>
<td>CBLHEAD</td>
</tr>
<tr>
<td>VPassword='hexstring'</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>+X'00'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td>Value 1</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>VDateRange</td>
<td>Date Range</td>
<td>-</td>
</tr>
<tr>
<td>INamDsn</td>
<td>File ID</td>
<td>-</td>
</tr>
<tr>
<td>PageDepth</td>
<td>Page Depth</td>
<td>CBLCLINE</td>
</tr>
<tr>
<td>PageWidth</td>
<td>Page Width</td>
<td>CBLCPW</td>
</tr>
<tr>
<td>Separator</td>
<td>Character</td>
<td>CBLCSEP</td>
</tr>
<tr>
<td>VLineSpace</td>
<td>Line Space</td>
<td>CBLVCSPA</td>
</tr>
<tr>
<td>VAssoc</td>
<td>Association</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VFlagImbed</td>
<td>Flag Imbed</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VFlagRepI</td>
<td>Flag Replicate</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VVolInfo</td>
<td>Vol Info</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VAlias</td>
<td>Alias</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VShrOrSC</td>
<td>Share or Share</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VPCOrUnused</td>
<td>PC Or Unused</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VTSOrDefined</td>
<td>TS Or Defined</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VExcpOrCICA</td>
<td>Exclusive or CICA</td>
<td>CBLVCWS1</td>
</tr>
<tr>
<td>VCI splits</td>
<td>CI Splits</td>
<td>CBLVCSCI</td>
</tr>
<tr>
<td>VCA splits</td>
<td>CA Splits</td>
<td>CBLVCSGA</td>
</tr>
<tr>
<td>VSecAlloc</td>
<td>Security Allocation</td>
<td>CBLVCALW</td>
</tr>
<tr>
<td>VSecAllocErr</td>
<td>Security Allocation Err</td>
<td>CBLVCAGE</td>
</tr>
<tr>
<td>VFilePercent</td>
<td>File Percent</td>
<td>CBLVPCF</td>
</tr>
<tr>
<td>VVolsPercent</td>
<td>VVol Percent</td>
<td>CBLVCPCV</td>
</tr>
<tr>
<td>VCatFileName</td>
<td>CBLVCFSN</td>
<td>-</td>
</tr>
<tr>
<td>VVolSer</td>
<td>Volume Serial</td>
<td>CBLVCWS2</td>
</tr>
<tr>
<td>VFreeSpace</td>
<td>Free Space</td>
<td>CBLVCWS2</td>
</tr>
<tr>
<td>VFreeTab</td>
<td>Free Tab</td>
<td>CBLVCWS2</td>
</tr>
<tr>
<td>VExpd</td>
<td>Expanded</td>
<td>CBLVCWS2</td>
</tr>
<tr>
<td>VPerm</td>
<td>Perm</td>
<td>CBLVCWS2</td>
</tr>
<tr>
<td>VMountVVDs</td>
<td>Mount VVD</td>
<td>CBLVCWS3</td>
</tr>
<tr>
<td>VOlay</td>
<td>Overlay</td>
<td>CBLVCWS3</td>
</tr>
<tr>
<td>VIndex</td>
<td>Index</td>
<td>CBLVCWS3</td>
</tr>
<tr>
<td>VAvlrOrLmax</td>
<td>AVL or Lmax</td>
<td>CBLVCWS3</td>
</tr>
<tr>
<td>VTunePrtSev</td>
<td>Tune Print Service</td>
<td>CBLVCWS3</td>
</tr>
<tr>
<td>VTunePrtCap</td>
<td>Tune Print Capacity</td>
<td>CBLVCWS3</td>
</tr>
<tr>
<td>VTunePrtJCL</td>
<td>Tune Print JCL</td>
<td>CBLVCWS3</td>
</tr>
<tr>
<td>VTuneDFP22</td>
<td>Tune DF P22</td>
<td>CBLVCWS3</td>
</tr>
<tr>
<td>VCrdInput</td>
<td>Card Input</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VNonICF</td>
<td>Non ICF</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VJCLBufnd</td>
<td>JCL Buffer</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VForceCancel</td>
<td>Force Cancel</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VOpsMsg</td>
<td>Operation Message</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VOnline</td>
<td>Online</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VCIICS</td>
<td>CIICS</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VIMS</td>
<td>IMS</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VDLI</td>
<td>DLI</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VDL1</td>
<td>DLI 1</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VString</td>
<td>String</td>
<td>CBLVCWS4</td>
</tr>
<tr>
<td>VOsTr</td>
<td>OStr</td>
<td>CBLVCWNS</td>
</tr>
<tr>
<td>VMaxOnlineCl</td>
<td>Max Online CI</td>
<td>-</td>
</tr>
<tr>
<td>VSortOrder</td>
<td>Sort Order</td>
<td>CBLVCWS5</td>
</tr>
<tr>
<td>VMaxExtents</td>
<td>Max Extents</td>
<td>CBLVCNX</td>
</tr>
<tr>
<td>VMinRC</td>
<td>Min RC</td>
<td>CBLVCRM</td>
</tr>
<tr>
<td>VVTOCPercnt</td>
<td>VTOC Percent</td>
<td>CBLVCPC</td>
</tr>
<tr>
<td>VDefine</td>
<td>Define</td>
<td>CBLVCWS6</td>
</tr>
</tbody>
</table>

2013-10-29 15:43:01 CBLVCAT Manual 136
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Offset</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGDGRepeat</td>
<td>CBLVCW6</td>
<td>X'40' +X'98'</td>
</tr>
<tr>
<td>VGDGDataset</td>
<td>CBLVCW6</td>
<td>X'20' +X'98'</td>
</tr>
<tr>
<td>VDefCatName</td>
<td>CBLVCW6</td>
<td>X'10' +X'98'</td>
</tr>
<tr>
<td>VReorg</td>
<td>CBLVCW6</td>
<td>X'08' +X'98'</td>
</tr>
<tr>
<td>VReorgDisk</td>
<td>CBLVCW6</td>
<td>X'04' +X'98'</td>
</tr>
<tr>
<td>VUseCatDSN</td>
<td>CBLVCW6</td>
<td>X'02' +X'98'</td>
</tr>
<tr>
<td>VDefPath</td>
<td>CBLVCW7</td>
<td>X'80' +X'99'</td>
</tr>
<tr>
<td>VDefBldIndex</td>
<td>CBLVCW7</td>
<td>X'40' +X'99'</td>
</tr>
<tr>
<td>VDefComments</td>
<td>CBLVCW7</td>
<td>X'20' +X'99'</td>
</tr>
<tr>
<td>VDefNotes</td>
<td>CBLVCW7</td>
<td>X'10' +X'99'</td>
</tr>
<tr>
<td>VDefCluster</td>
<td>CBLVCW7</td>
<td>X'08' +X'99'</td>
</tr>
<tr>
<td>VDefAix</td>
<td>CBLVCW7</td>
<td>X'04' +X'99'</td>
</tr>
<tr>
<td>VDefComment</td>
<td>CBLVCW7</td>
<td>X'02' +X'99'</td>
</tr>
<tr>
<td>VDeJCL</td>
<td>CBLVCW7</td>
<td>X'01' +X'99'</td>
</tr>
<tr>
<td>VDelCatName</td>
<td>CBLVCW8</td>
<td>X'10' +X'9A'</td>
</tr>
<tr>
<td>VISCatalog</td>
<td>CBLVCW9</td>
<td>X'04' +X'9B'</td>
</tr>
<tr>
<td>VTZAdjust</td>
<td>CBLVCW9</td>
<td>X'02' +X'9B'</td>
</tr>
<tr>
<td>V2digitYear</td>
<td>CBLVCW9</td>
<td>X'01' +X'9B'</td>
</tr>
</tbody>
</table>

Full CBLNAME field descriptions follow in CBLNAME offset order. Each field name is displayed together with details of its offset in CBLNAME, its length and also its default value. All default values are those as supplied in the file as distributed.

### Detailed Field Descriptions

<table>
<thead>
<tr>
<th>Offset</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>+X'00'</td>
<td>CBLHEAD DC CL55&quot;Your Installation Name - Location&quot;</td>
</tr>
<tr>
<td></td>
<td>This field contains your installation or company name exactly as supplied by CBL (case sensitive with blanks, commas, etc. preserved). It is mandatory that this field be specified. The site name distributed to all users is restricted to between 20 and 36 characters in length. Since the CBLHEAD field can accommodate up to 55 characters, different headings may still be achieved by varying the content of positions 37 to 55. Note, however, that SELCOPY CMS users should not do this since the first 16 characters of the last command are displayed at CBLHEAD+38.</td>
</tr>
<tr>
<td>+X'37'</td>
<td>CBLNAMV DC X'00' Release no. of this CBLNAME module.</td>
</tr>
<tr>
<td></td>
<td>This field must contain X'00'.</td>
</tr>
<tr>
<td>+X'38'</td>
<td>CBLCLINE DC AL1(0) Default page depth.</td>
</tr>
<tr>
<td></td>
<td>The lines printed per page may vary between 8 and 255, i.e. AL1(8) to AL1(255). If the value is left at 0, 58 lines per page will be used for MVS systems, and the SYSLST system default will be used for VSE. Run-time override is available via PAGEDEPTH=nn (or CBLCLINE=nn).</td>
</tr>
<tr>
<td>+X'39'</td>
<td>CBLCPW DC AL1(0) Default heading width.</td>
</tr>
<tr>
<td></td>
<td>The heading width may vary between 72 and 160. If the value is left at 0, a heading width of 132 will be used for standard reports. For REPORT VCAT/VTOC the heading width is implicitly defined by the REPORT operation. Run-time override is available via PAGEWIDTH=nn.</td>
</tr>
<tr>
<td>+X'3A'</td>
<td>DC XL2(0) Reserved.</td>
</tr>
<tr>
<td></td>
<td>This field must contain zeros, for upward compatibility with future versions of CBL products.</td>
</tr>
<tr>
<td>+X'3C'</td>
<td>CBLCCTL DC CL2&quot;.&quot; Control characters.</td>
</tr>
<tr>
<td>+X'3E'</td>
<td>CBLCSEP DC CL1&quot;!&quot; Separator character.</td>
</tr>
<tr>
<td>+X'3F'</td>
<td>DC XL17(0) Reserved.</td>
</tr>
<tr>
<td></td>
<td>This field must contain zeros, for upward compatibility with future versions of CBL products.</td>
</tr>
<tr>
<td>Non-CBLVCAT Fields</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td></td>
</tr>
<tr>
<td>+X'50' to +X'67'</td>
<td>SELCOPY CBLNAME fields.</td>
</tr>
<tr>
<td>+X'68' to +X'73'</td>
<td>SELUPD CBLNAME fields, (Obsolete Product)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CBLVCAT Specific Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>+X'74'</td>
</tr>
<tr>
<td><strong>CBLVCSPA DC X'02'</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

| +X'75'                  |
| **CBLVCSW1 DC X'FF'**   | LISTVCAT options (part 1). |
|                         | This byte contains 8 independent switches that control various aspects of the LISTVCAT report (Set the byte by adding the required values together, e.g. X'80'+X'20'+X'04' = X'A4'). |
|                         | Unlike the other switch fields that follow, CBLVCSW1 defaults to X'FF', with all bits on. See CBLVCSW3 and CBLVCSW6 for other LISTVCAT options. |
| **X'80'** ON = Sets Option ASSOC. |
| Association details can be displayed (ASSOC) or suppressed (NOASSOC). |
| **X'40'** ON = Display ---IMB--- eyecatcher. |
| Files with the IMBED attribute can have the eyecatcher ---IMB--- displayed in the KL,RKP/BLK/IMB column. |
| **X'20'** ON = Display ---REP--- eyecatcher. |
| Files with the REPLICATE attribute can have the eyecatcher ---REP--- displayed in the KL,RKP/BLK/IMB column. |
| **X'10'** ON = Sets Option VOLINFO. |
| Volume information can be displayed or suppressed. |
| **X'80'** ON = Sets Option ALIAS. (For MVS) |
| Alias information can be displayed (ALIAS) or suppressed (NOALIAS). |
| **X'80'** ON = Sets Option SHR. (For VSE) |
| Display either the share options only (SHR) or the class and the share options (S/C). |
| **X'04'** ON = Sets Option PCNT. |
| Display either the percentage used (PCNT) or number of unused (UNUSED) tracks/blocks of a file's allocated space. |
| **X'02'** ON = Sets Option TIMESTMP. |
| Display either the timestamp (TIMESTMP) or the defined date (DEFINED). |
| **X'01'** ON = Sets Option EXCPS. |
| Display either the number of Execute Channel Program operations (EXCPS) or the number of control intervals per control area (CI/CA). |

| +X'76'                  |
| **CBLVCSCI DC AL1(1)** | CI splits PCNT threshold. |
| When the number of CI Splits per 100 inserts to a file is greater than or equal to the value in this field, then one of the messages SEV 2-08 or SEV 3-08 "CI SPLITS TOO HIGH" is displayed (the SEVerity level depends upon the value in this field). |
| However, if the total number of inserts in a file is less than 100, then one of the above SEVerity messages is displayed when the actual number of CI splits is greater than or equal to the value in this field. |
| The allowable values range from 1 to 99 inclusive. |

| +X'77'                  |
| **CBLVCSCA DC AL1(1)** | CA splits PCNT threshold. |
| When the number of CA Splits per 100 inserts to a file is greater than or equal to the value in this field, the message SEV 3-06 "CA SPLITS TOO HIGH" is displayed. |
| If there are CA splits and the threshold is not reached, the message SEV 2-05 "CA SPLITS EXIST" will be printed instead. |
| The allowable values range from 1 to 99 inclusive. |

| +X'78'                  |
| **CBLVCALW DC AL1(1)** | The warning threshold for Secondary Extents. |
| This LISTVCAT field contains the warning threshold for files acquiring secondary extents. |
This affects the format of the displayed **ALLOCS** value and the setting of SEV 1-18, SEV 2-19 or SEV 3-19.

If the **CBLVCALW** threshold (but not the **CBLVCALE** threshold) is reached, the SEV 2-19 message "**SEC EXTENTS TOO HIGH**" is printed.

The displayed **ALLOCS** value will have "**n**" appended, where 'n' is the number of secondary extents.

If Secondary extents exist but the threshold is not reached, the message SEV 1-18 "**SEC EXTENTS EXIST**" is displayed.

### **CBLVCALE DC AL1(10)**

The error threshold for Secondary Extents.

This **LISTVCAT** field contains the error threshold for files acquiring secondary extents.

If the number of secondary extents reaches this value, the SEV 2-19 message "**SEC EXTENTS TOO HIGH**" is printed.

The displayed **ALLOCS** value will have "****" appended and (for the standard report only) "**** nnn SEC EXTENTS ****" (showing the number of extents) will appear on the following line.

See the **CBLVCALW** threshold) is field for the action taken when secondary allocation is less than **CBLVCALE**.

### **CBLVCPGF DC AL1(85)**

File PCNT full threshold.

This **LISTVCAT** field contains the file full percentage threshold.

Files that meet or exceed this value will show "**nn.n**" in the PCNT field of the detail report (The **PCNT** option is required to print this field on the standard report).

This value is also used to trigger the SEV 1-09 message "**FILE GETTING/IS FULL**"

### **CBLVCPCV DC AL1(85)**

Volume PCNT threshold.

This **LISTVCAT** field contains the percentage full threshold for the space controlled by VSAM catalogs.

When the volume summary is printed, a statistics line is also printed for each data space controlled by the catalog.

If the percentage of used space meets or exceeds this value, it is displayed as "**nn.n**":

### **CBLVCFN DC XL8'0' VSAM Catalog name.**

This **LISTVCAT** field holds the VSAM catalog filename (DDNAME) used if the **DDNAME** parameter is omitted on the first **LISTVCAT** operation.

Subsequent **LISTVCAT** operations for the run will default to catalog previously specified.

If the field is unchanged (contains binary zeros), then the catalog name defaults to IJSYSUC.

### **CBLVCSW2 DC X'00'**

**LISTVTOC** options.

This byte contains 8 independent switches that control various aspects of the **LISTVTOC** report (Set the byte by adding the required values together, e.g. X'80'+X'20'+X'04' = X'A4').

- **X'00'** Reserved.
  These bits are reserved and should be set off.
- **X'20'** ON = Suppress display of volume serial number.
  The volume serial number can be displayed if it differs from the current volume (On some **MVS** systems, the volume serial number field in the DSCB1 (Format 1) record may be used for other purposes).
- **X'10'** ON = Sets Option **FREE/FREETAB**.
  Freespace information is to be displayed (**FREE** or **FREETAB**) or not (**NOFREE**), the format of the freespace information depends on the **X'08'** bit. (See below).
- **X'08'** ON = Sets Option **FREETAB**.
  If the **X'10'** bit is on, this controls the way in which freespace is displayed. Freespace can either be displayed in a table format (**FREETAB**) or within the main report (**FREE**).
- **X'04'** ON = Sets Option **NOEXPD**.
  The user can choose to display **EXPD** (**EXPD**), instead of the date (**NOEXPD**), for any files past their expiry date.
- **X'02'** ON = Sets Option **NOPERM**.
  The user can choose to display **PERMANENT** (**PERM**), instead of the date (**NOPERM**), for any files with an expiry date equal to 1999/365.
<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'01'</td>
<td>Reserved. This bit is reserved and should be set off.</td>
</tr>
<tr>
<td>+X'85'</td>
<td>CBLVCSW3 DC X'00' LISTVCAT options (part 2). This byte contains a further 8 independent switches that control various aspects of the LISTVCAT report. See CBLVCSW1 and CBLVCSW6 for other LISTVCAT options.</td>
</tr>
<tr>
<td>X'80'</td>
<td>ON = Sets Option NOMOUNT. For ICF catalogs only, controls whether the system operator will be asked to mount off-line VVDS volumes (Any file that refers to a non-mounted volume will display the message &quot;VVDS NOT OPEN&quot;).</td>
</tr>
<tr>
<td>X'40'</td>
<td>ON = Sets Option OVLAY. Filenames longer than the DSN report field either overwrite the following fields or the filename is printed immediately, and all other fields are printed on the following line.</td>
</tr>
<tr>
<td>X'20'</td>
<td>ON = Sets Option NOINDEX. For KSDS/AIX, the file may use two lines to display the data and index components, or it may be restricted to the data component only.</td>
</tr>
<tr>
<td>X'10'</td>
<td>ON = Sets Option AVRL. Choose either the AVRL or LMAX field on the LISTVCAT standard report.</td>
</tr>
<tr>
<td>X'08'</td>
<td>ON = Sets Option NOPSEV. Print or suppress the SEVerity messages during tuning runs. This switch does not affect subset SEV=n if there is no tuning specified.</td>
</tr>
<tr>
<td>X'04'</td>
<td>ON = Sets Option NOPCAP. Print or suppress the tuning Capacity block.</td>
</tr>
<tr>
<td>X'02'</td>
<td>ON = Sets Option NOPJCL. Print or suppress the JCL Override Block for sequential processing.</td>
</tr>
<tr>
<td>X'01'</td>
<td>ON = TUNE for DFP prior to 2.2 (MVS only). When this bit is on, tuning recommendations are based on pre DFP 2.2 system considerations.</td>
</tr>
<tr>
<td>+X'86'</td>
<td>CBLVCSW4 DC X'00' Release dependencies. This byte contains a further 8 independent switches that control various general release dependencies.</td>
</tr>
<tr>
<td>X'80'</td>
<td>ON = The CBLVCAT and CBLVTOC programs do not use SYSIPT/SYSIN. This bit maintains compatibility with job control produced for previous releases of CBLVCAT (CBLVCAT and CBLVTOC, together with CBLVTOCM, are no longer maintained, or distributed, by CBL). It is recommended that CBLV is used as the invocation program (VSE/ESA can only use CBLV).</td>
</tr>
<tr>
<td>X'40'</td>
<td>ON = Use the DFP 3.1 method to avoid STEPCATS. STEPCAT/JOBCAT switch for LISTVCAT. This bit does not affect VSE systems and is no longer used for MVS systems as STEPCATs for ICF were made redundant in DFP 3.1. THIS BIT IS SET BY DEFAULT AT RUN-TIME, UNLESS A SPECIAL ZAP HAS BEEN APPLIED. ICF users with levels of DFP below 3.1 who run without STEPCATs should contact CBL to revert to the old (Rel 9.2) error prone method of dynamically allocating the catalog.</td>
</tr>
<tr>
<td>X'20'</td>
<td>ON = VM MVS non-ICF catalog. VM users accessing MVS non-ICF catalogs, require this bit on for the DEFINE parameter.</td>
</tr>
<tr>
<td>X'10'</td>
<td>Reserved. This bit is reserved and should be set off.</td>
</tr>
<tr>
<td>X'08'</td>
<td>ON = BUFND in JCL block. When this bit is on, BUFND information will be included in the tuning JCL BLOCK for users prior to VSE/SP 4.1 (This can be of use when modelling for a new VSE release).</td>
</tr>
<tr>
<td>X'04'</td>
<td>ON = Force Cancel for VSE. For VSE/SP 2 and higher, with this bit on, a CANCEL is forced if appropriate, instead of passing a return code. Thus catering for OEM software that does not recognise OPERATING SYSTEM Return Codes.</td>
</tr>
<tr>
<td>X'03'</td>
<td>ON = Suppress console messages. When these bits are on, console messages of control statement and run time errors are suppressed.</td>
</tr>
<tr>
<td>+X'87'</td>
<td>CBLVCONL DC X'00' ONLINE file control switch.</td>
</tr>
</tbody>
</table>
This field controls which "character strings", within the filename, are to be used to identify on-line files (Set the byte by adding the required values together, e.g. X'80'+X'20'+X'04' = X'A4').

The string must be a complete segment for a match to occur.

If the match is successful, the file will be considered to be eligible for on-line use, and will be tuned accordingly. The strings supported are:

- X'80' ON = 'ONLINE'
- X'40' ON = 'CICS'
- X'20' ON = 'IMS'
- X'10' ON = 'DLI'
- X'08' ON = 'DL1'
- X'06' ON = Reserved.
  These bits are reserved, and should be set off.
- X'01' ON = The "string" contained in CBLVCONS.

**+X'88'** CBLVCONS DC XL8'0' ONLINE file match string.

If field CBLVCONL (above) has X'01' set on, CBLVCAT will check the data set name for a match with the contents of this field.

If the match is successful, the file will be considered to be eligible for on-line use, and will be tuned accordingly.

Specify: **CBLVCONS DC CL8'string'** to change this field.

**+X'90'** CBLVCONT DC F'0' ONLINE maximum CI size.

This is the CIMAX value used if the file has been identified as on-line.

If this field is set to zero, a CI size of 4096 is used.

Specify: **CBLVCONT DC F'nnnn'** where nnnn is any valid CI size you wish to enforce.

**+X'94'** CBLVCSW5 DC X'00' LISTVTOC Sort order.

Supply one of the following values, in order to select the file sequence within VTOC reports (This order may be changed at run-time via the SORT parameter):

- X'A0' ON = By SIZE (descending)
- X'90' ON = By DATE (descending)
- X'04' ON = By EXP.
- X'02' ON = By DSN.
- X'01' ON = By EXTPRIME.
- X'00' ON = By EXTENT.

**+X'95'** CBLVCEXT DC X'00' LISTVCAT maximum number of extents printed.

This field should normally be set to X'00', as the value is designed to be set on at run time using option CBLVCEXT=nn. This will display up to n physical extents per file and can be useful for disk performance monitoring.

**+X'96'** CBLVCRCM DC X'00' Minimum Return Code.

The minimum reported Return Code (Values below this minimum are suppressed).

Specify: **CBLVCRCM DC X'10'** to suppress Return Codes 1 to 15.

**+X'97'** CBLVCPCT DC AL1(85) VTOC percentage tracks used (PCNT) threshold.

This LISTVTOC field contains the percentage tracks used threshold.
When a LISTVTOC with FREETAB report is requested, the summary includes a line identified as TOTAL FREE TRACKS which also includes the percentage used. If this percentage used meets or exceeds this value, it is displayed as "**nn.n**".

**LISTVCAT options (part 3).**

This byte contains a further 8 independent switches that control various aspects of the LISTVCAT report (Set the byte by adding the required values together, e.g. X'80' + X'20' + X'04' = X'A4').

See CBLVCSW1 and CBLVCSW3 for other LISTVCAT options.

- **X'80'** ON = Sets Option DEFINE. Produces IDCAMS DEFINE deck output (See also CBLVCSW7 and CBLVCSW8 settings for related switches).

- **X'40'** ON = Sets Option GDGRPT. The GDG name is repeated on every line of a customised report which contains overflow information for that file.

- **X'20'** ON = DSN with generation number. NOINDEX. Append the GDG Dataset with the absolute generation number.

- **X'10'** ON = Suppress Catalog name. The DEFINE deck can include the Catalog name statement. If excluded the define defaults to the 'Order of Catalog Selection'.

- **X'08'** ON = Create Reorganisation deck. The reorganisation facility produces a skeleton jobstream which requires user attention. A suitable warning message, WARN 019, is included in the jobstream, before the IDCAMS DELETE, to emphasise this point (It is assumed that the CBL product SELCOPY is to be used to perform the file back up/restore operation). See the X'04' setting for further information.

- **X'04'** ON = Use disk for Reorganisation work file. Reorganisation (see above) defaults to tape as the backup media, this can be altered to disk by setting this switch on.

- **X'02'** ON = Catalog name. Use Catalog name, instead of DDNAME, for heading of the DSN column.

- **X'01'** Reserved. This bit is reserved and should be set off.

**LISTVCAT DEFINE defaults.**

This byte contains a further 8 independent switches that control various aspects of the LISTVCAT DEFINE parameter production. (Set the byte by adding the required values together, e.g. X'80' + X'20' + X'04' = X'A4').

See CBLVCSW6 and CBLVCSW8 for other LISTVCAT DEFINE settings.

- **X'80'** ON = PATH parameters suppressed. Path decks selected by the specified SUBSET parameter(s) can be filtered out by setting this switch on.

- **X'40'** ON = BLDINDEX parameters suppressed. BLDINDEX constructs the data in the Alternate Index dataset. The Prime and Alternate keys from the Base Cluster are used to compose the data portion of the AIX (The BLDINDEX deck, if requested, follows the DEFINE AIX deck).

- **X'20'** ON = Suppress DEFINE comments. The old define values are included in a 'commented out' form in the define deck. This option allows the user to suppress the production of these comments.

- **X'10'** ON = Include Notes. The tuning notes available in the SYSLST report can also be part of the SYSPCH output.

- **X'08'** ON = Cluster parameters suppressed. Cluster decks selected by the specified SUBSET parameter(s) can be filtered out by setting this switch on.

- **X'04'** ON = AIX parameters suppressed. AIX parameters selected by the specified SUBSET parameter(s) can be filtered out by setting this switch on.

- **X'04'** ON = DELETE parameters not commented. The DELETE parameters preceeding the requested DEFINE parameters can be "commented out" ('/*' in positions 2 and 3).

- **X'01'** ON = Job control suppressed. MVS/VSE job control may be wrapped around the DEFINE parameters.
### +X'9A'
<table>
<thead>
<tr>
<th>CBLVCSW8 DC X'00'</th>
<th>LISTVCAT IDCAMS DELETE defaults.</th>
</tr>
</thead>
<tbody>
<tr>
<td>This byte contains a single switch that controls IDCAMS DELETE parameter production.</td>
<td></td>
</tr>
<tr>
<td>X'10' ON = Suppress Catalog name.</td>
<td></td>
</tr>
<tr>
<td>The <strong>DELETE</strong> parameters can include the Catalog name statement. If excluded it defaults to the 'Order of Catalog Selection'. Suppression of the Catalog name should be approached with care.</td>
<td></td>
</tr>
</tbody>
</table>

### +X'9B'
<table>
<thead>
<tr>
<th>CBLVCSW9 DC X'00'</th>
<th>General CBLVCAT defaults.</th>
</tr>
</thead>
<tbody>
<tr>
<td>This byte contains a further 8 independent switches that control various aspects of the <strong>LISTVCAT</strong> and <strong>LISTVTOC</strong> report. (Set the byte by adding the required values together, e.g. X'04'+X'02'+X'01' = X'07').</td>
<td></td>
</tr>
<tr>
<td>X'10' ON = Suppress <strong>X</strong> ovetype in <strong>LISTVCAT</strong> <strong>TYPE</strong> field.</td>
<td></td>
</tr>
<tr>
<td>The <strong>REPORT VCAT</strong> parameter <strong>TYPE</strong> (included as part of a standard <strong>LISTVCAT</strong> report), displays the data set type based on its DSORG, RECORG, etc.</td>
<td></td>
</tr>
<tr>
<td><strong>CBLVCAT</strong> overwrites the 4th byte of the type field with 'X' if an MVS VSAM data set is defined with Extended Addressability or if a VSE VSAM cluster is defined with <strong>EXTRALARGEDATASET</strong>. Therefore, a KSDS data set with Extended Addressability will be reported as type KSDX, an ESDS data set with Extended Addressability will be reported as type ESDX, etc.</td>
<td></td>
</tr>
<tr>
<td>Setting this bit on suppress this overwrite.</td>
<td></td>
</tr>
<tr>
<td>X'08' ON = Display Hexadecimal representation of devices.</td>
<td></td>
</tr>
<tr>
<td><strong>REPORT VCAT</strong> parameter <strong>VOL2/4/5</strong> for customised reports, prints the <strong>DEVICE</strong> column indicating the device type of the volume on which extents exist for a non-VSAM data set.</td>
<td></td>
</tr>
<tr>
<td>Setting this bit on forces hexadecimal representation of the device type.</td>
<td></td>
</tr>
<tr>
<td>X'04' ON = VSAM In-Storage Catalog Processing</td>
<td></td>
</tr>
<tr>
<td>Sets option <strong>ISC</strong> to improve performance of VSAM catalog reporting for catalogs containing a large number of entries.</td>
<td></td>
</tr>
<tr>
<td>Even if this bit is set ON, <strong>CBLVCAT</strong> will not invoke In-Storage Catalog processing if <strong>KEY</strong>= and/or <strong>IGN</strong>= subsetting is specified. To override this, for instances when <strong>KEY</strong>= subsetting results in a significant proportion of a large catalog being selected, the run-time option <strong>ISC</strong> should be explicitly specified in the control statements.</td>
<td></td>
</tr>
<tr>
<td>X'02' ON = Catalog Report with local <strong>TIMESTMP</strong> date field.</td>
<td></td>
</tr>
<tr>
<td>Adjust the <strong>LISTVCAT</strong> <strong>TIMESTMP</strong> column date and time using local <strong>TIMEZONE</strong> (+ or - displacement from GMT).</td>
<td></td>
</tr>
<tr>
<td>Timestamp information is stored in a VSAM and ICF catalog as a Greenwich Mean Time (GMT) value. This value is adjusted by the value stored in the system local offset (Zone) controlled locally as follows:</td>
<td></td>
</tr>
<tr>
<td>- <strong>MVS</strong> - <strong>TIMEZONE</strong> W,mm,ss in <strong>SYS1.PARMLIB(CLOCKnn)</strong>.</td>
<td></td>
</tr>
<tr>
<td>- <strong>VSE</strong> - <strong>IPL Command</strong> <strong>SET ZONE=W,HH/MM</strong>.</td>
<td></td>
</tr>
<tr>
<td>- <strong>VM</strong> - <strong>TIMEZONE DEFINITION</strong> in <strong>SYSTEM CONFIG</strong>.</td>
<td></td>
</tr>
<tr>
<td>Note that ICF and VSAM catalogs hold Timestamp information in binary units of 1.048576 seconds. The local offset in MVS and VM is also held in binary seconds. However, for VSE systems the local offset is stored in minutes, therefore the accuracy of the seconds value in the reported Local <strong>TIMESTMP</strong> column is subject to rounding errors.</td>
<td></td>
</tr>
<tr>
<td>X'01' ON = Reports with 2 digits years in date fields.</td>
<td></td>
</tr>
<tr>
<td>Catalog report columns <strong>TIMESTMP</strong> and <strong>DEFINED</strong> display 2 digit year instead of 4 digit year. Also, <strong>S/C</strong> or <strong>SHR</strong> is included as part of the standard <strong>LISTVCAT</strong> report. <strong>VTOC</strong> report columns <strong>ACCESSED</strong>, <strong>CREATED</strong> and <strong>EXPIRES</strong> display 2 digit year instead of 4 digit year.</td>
<td></td>
</tr>
</tbody>
</table>

### More Non-CBLVCAT Fields
<table>
<thead>
<tr>
<th>+X'A0' to +X'BF'</th>
<th>CBLDO CBLNAME fields. (Obsolete Product)</th>
</tr>
</thead>
<tbody>
<tr>
<td>+X'9A'</td>
<td>DC XL64'0'</td>
</tr>
<tr>
<td>This field must contain zeros, for upward compatibility with future versions of CBL products.</td>
<td></td>
</tr>
</tbody>
</table>
The **QUERY CBLNAME** operation produces a formatted display of the fields within **CBLNAME** that influence **CBLVCAT**. Any previously supplied **OPTION** overrides will affect the output.

```
| +00 START OF CBLNAME |
| +38 CBLCLINE = X'56'   | PAGEDEPTH |
| +74 CBLVCSPA = X'01'   | LINE SPACING |
| +75 CBLVCSW1 = X'FE'   | VCAT OPTIONS 1 |
| +76 CBLVCS1 = X'01'    | FLAG THRESHOLD - CI SPLITS |
| +77 CBLVCSCA = X'01'   | FLAG THRESHOLD - CA SPLITS |
| +78 CBLVCALM = X'04'   | FLAG THRESHOLD - SEC ALLOC |
| +79 CBLVCALF = X'0A'   | ERROR THRESHOLD - SEC ALLOC |
| +7A CBLVCPCF = X'55'   | FLAG THRESHOLD - FILES PCNT |
| +7B CBLVCPCV = X'55'   | FLAG THRESHOLD - VOLS PCNT |
| +7C CBLVCPW2 = X'50'   | VSAM CATALOG NAME |
| +84 CBLVCSPW2 = X'00'  | VCAT OPTIONS |
| +85 CBLVCSW3 = X'00'   | VCAT OPTIONS 2 |
| +86 CBLVCSW4 = X'00'   | VCAT/VTOC SWITCHES |
| +87 CBLVCONL = X'00'   | ONLINE SWITCH FOR VCAT TUNING |
| +88 CBLVCNLS = ' '     | STRING USED BY CBLVCNL |
| +90 CBLVCPN = X'00000100' | MAX CISIZE FOR ONLINE TUNING |
| +95 CBLVCXCT = X'00'   | VCAT MAX EXTENTS TO PRINT |
| +96 CBLVCXCM = X'00'   | MIN RETURN CODE REQUIRED |
| +97 CBLVCXCM = X'00'   | FLAG THRESHOLD - VTOC PCNT |
| +98 CBLVCSW6 = X'00'   | VCAT OPTIONS 3 |
| +99 CBLVCSPW7 = X'00'  | VCAT DEFINE OPTIONS |
| +9A CBLVCSPW8 = X'00'  | VCAT DELETE OPTION |

** ** ** ** ** ** ** CBLVCAT IS LICENSED BY COMPUTE (BRIDGEND) LTD 06/1995

** EXPIRY DATE --- 6 JUN 1995
```

Example: **QUERY CBLNAME** (CBL Ref: vmxqnam)
Messages

SEV Messages (Severity)

This section describes the SEVerality messages that are displayed during subset SEV=n runs and also during tuning runs. These messages are CBLVCAT's method of indicating a file's state of tune.

e.g.

*** SEV 3-19 *** SEC EXTENTS TOO HIGH

This example indicates that a SEV=3 (major) problem has been encountered, which is described under SEV message No. 19.

When a file has been selected (either because it is out of tune for a tuning run, or by a subset SEV=n run) all the severity messages which apply to that file are displayed.

For all the SEV messages produced, CBLVCAT's TUNE operation will recommend the changes to the DEFINE parameters which will address the problem(s).

The following table lists all CBLVCAT’s severity messages:

<table>
<thead>
<tr>
<th>No.</th>
<th>SEV Levels</th>
<th>Message Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>3 2</td>
<td>ACTUAL FRSP CA = nn PC</td>
</tr>
<tr>
<td>02</td>
<td>3 2</td>
<td>ACTUAL FRSP CI = nn PC</td>
</tr>
<tr>
<td>03</td>
<td>3</td>
<td>ACTUAL FRSP CI = 0 PC</td>
</tr>
<tr>
<td>04</td>
<td>- 2</td>
<td>BUFSP TOO SMALL FOR EFFICIENCY</td>
</tr>
<tr>
<td>05</td>
<td>- 2</td>
<td>CA SPLITS EXIST</td>
</tr>
<tr>
<td>06</td>
<td>3</td>
<td>CA SPLITS TOO HIGH (nn PC OF INSERTS)</td>
</tr>
<tr>
<td>07</td>
<td>-</td>
<td>CI SPLITS EXIST</td>
</tr>
<tr>
<td>08</td>
<td>3 2</td>
<td>CI SPLITS TOO HIGH (nn PC OF INSERTS)</td>
</tr>
<tr>
<td>09</td>
<td>-</td>
<td>FILE GETTING/IS FULL</td>
</tr>
<tr>
<td>10</td>
<td>3 2 1</td>
<td>IMBED COSTS nn PC OF DATA SPACE</td>
</tr>
<tr>
<td>11</td>
<td>- 2</td>
<td>INDEX CISIZE IS EXCESSIVE</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>INDEX EXCPS DWARF DATA</td>
</tr>
<tr>
<td>13</td>
<td>- 2</td>
<td>INDEX EXCPS EXCEED DATA</td>
</tr>
<tr>
<td>14</td>
<td>-</td>
<td>KSIDS/AIX HAS NO IMDEDDED FREE SPACE</td>
</tr>
<tr>
<td>15</td>
<td>- 2 1</td>
<td>nn CYLS CAN BE RECOVERED WHEN TUNED</td>
</tr>
<tr>
<td>16</td>
<td>-</td>
<td>NRECS IS ESTIMATED</td>
</tr>
<tr>
<td>17</td>
<td>-</td>
<td>SEC ALLOC DEFINED AS ZERO</td>
</tr>
<tr>
<td>18</td>
<td>-</td>
<td>SEC EXTENTS EXIST</td>
</tr>
<tr>
<td>19</td>
<td>3 2</td>
<td>SEC EXTENTS TOO HIGH</td>
</tr>
<tr>
<td>20</td>
<td>-</td>
<td>SHR 4 IS EXPENSIVE</td>
</tr>
<tr>
<td>21</td>
<td>-</td>
<td>SPANNED RECORDS</td>
</tr>
<tr>
<td>22</td>
<td>-</td>
<td>SPEED NOT DEFINED - RECOVERY IS DEFAULT</td>
</tr>
<tr>
<td>23</td>
<td>3</td>
<td>TIMESTAMP CONFLICT</td>
</tr>
<tr>
<td>24</td>
<td>-</td>
<td>WRITECHECK IS EXPENSIVE</td>
</tr>
<tr>
<td>25</td>
<td>- 2</td>
<td>INEFFICIENT DATA CISIZE</td>
</tr>
<tr>
<td>26</td>
<td>- 2 1</td>
<td>nnnn CYLS OVER-ALLOCATED</td>
</tr>
<tr>
<td>27</td>
<td>- 2</td>
<td>TUNING FOR RECS/AVLRECL CHANGE REQUEST</td>
</tr>
<tr>
<td>28</td>
<td>- 2</td>
<td>TUNING FOR DEVICE CHANGE REQUEST</td>
</tr>
<tr>
<td>29</td>
<td>- 2</td>
<td>TUNING FOR GROWTH REQUEST</td>
</tr>
<tr>
<td>30</td>
<td>- 2</td>
<td>TUNING FOR CISIZE CHANGE REQUEST</td>
</tr>
<tr>
<td>31</td>
<td>-</td>
<td>AVLRECL: DEFINED=nnnnn~ ESTIMATED=nnnnn</td>
</tr>
<tr>
<td>32</td>
<td>3</td>
<td>FILE SHOULD NOT BE SPANNED</td>
</tr>
</tbody>
</table>

A full description of all SEV messages, in numerical order, now follows.
01. ACTUAL FRSP CA = nn PC
   SEV=2/3 only:
   Applies to KSDS/AIX files only. The effective percentage of free CIs per CA differs significantly from the defined value.
   ◊ SEV=3. The actual freespace value is at least 20% and also four times higher than defined (This indicates that the file will waste disk space and that sequential performance will suffer).
   ◊ SEV=2. The actual freespace value is at least 10 percent and also double that defined, or is at least 20% higher than that defined.

   The problem arises because any non-zero FRSP CA request reserves at least one free CI per CA. If CI/CA is small then the minimum effective percentage can be very high (i.e. 50 percent if 2 CIs per CA).

   The problem may be compounded by the IMBED option, which will also reduce the available CIs per CA.

   Action Required:
   The problem is caused by a poor combination of FRSP CA and CA size. Correct the primary and secondary allocation sizes or change the FSPC second value.

02. ACTUAL FRSP CI = nn PC
   SEV=2/3 only:
   Applies to KSDS/AIX files only. The effective percentage of Freespace/CI differs significantly from that defined.
   ◊ SEV=3. The actual freespace value is at least 20% and also four times higher than defined (This indicates that the file will waste disk space and the sequential performance will suffer).
   ◊ SEV=2. The actual freespace value is at least 10 percent and also double that defined, or is at least 20 percentage points higher than that defined.

   When distributed freespace is requested within control intervals, there must be enough bytes available for at least one record insertion. If there are only a few records per control interval (due to large records and/or a small CI size) the actual amount of CI freespace reserved can be larger than the value used to define the file (e.g. For a CI containing 2 records, the minimum non-zero CI freespace is approximately 50%).

   Action Required:
   The problem is caused by a poor combination of FRSP CI and CI size. Adjust the CI size to accommodate the freespace percentage, or adjust the percentage to reflect the bytes to be left free for future inserts.

03. ACTUAL FRSP CI = 0 PC
   SEV=3 only:
   The file has been defined with a non-zero FRSP CI value and the control interval is not large enough to reserve any usable freespace.

   For fixed length records, this occurs if there is insufficient room for two records (allowing for VSAM's 10 control bytes per CI) whereas for variable length records it occurs if the number of bytes reserved is smaller than the average record size.

   Action Required:
   The problem is caused by a poor combination of FRSP CI, Average record length and CI size.

04. BUFSSP TOO SMALL FOR EFFICIENCY
   SEV=2 only:
   The minimum BUFSSP value defined is insufficient for efficient random processing (If the SEV 3-12 or SEV 2-13 message is also displayed, the file's performance has already suffered).

   For KSDS and AIX files, the buffer space should be large enough for two data control intervals and one index control interval per index level. A small bufferspace will increase the number of index EXCPS (which should always be significantly lower than the number of data EXCPS).

   Action Required:
   Alter the defined bufferspace or override it in the JCL.

05. CA SPLITS EXIST
   SEV=2 only:
   Applies to KSDS/AIX files only. CA splits have taken place, but the number of splits is less than the CA split threshold CBLVCSCA.

   If this message is produced when the file being reported is in its worst state (just prior to backup/restore), the condition may be tolerable. If, however, there is more insert activity to take place it should be investigated.

   CA splits can cause severe system or terminal response degradation. While it is ultimately best not to suffer CA splits, it may be that the cost of eliminating them entirely is too high compared to the increased file space required when specifying a high CA freespace percentage.

   The prime area of concern, when evaluating a tolerable CA split level, is the amount and nature of insert activity on the file. Other file attributes such as CI freespace, or different size CIs may also help the situation.

   Action Required:
   None essential, however, performance will be improved if the file is re-defined and re-loaded. If inserts are likely to continue, you should increase FRSP CI and/or FRSP CA first.
06. CA SPLITS TOO HIGH (nn PC OF INSERTS)
SEV=2 only:
Applies to KSDS/AIX files only. The percentage of inserts causing CA splits exceeds the CBLVCSCA threshold.

When files consistently show this message it is an indication the file needs more frequent re-organisation or that inserts are occurring heavily in pockets around the file, rather than being evenly distributed.

When a Control Area split occurs, it can have a dramatic impact on performance. Although the degree of degradation depends on the mode of processing, the CA size and the amount of data being moved, it is ultimately desirable to avoid all CA splits.

If record insert activity occurs heavily in pockets, the obvious immediate solution would be to increase CA freespace which would require more space for the file. To compensate, it is worth considering a reduction in CI freespace since only small areas of the file are insert bound. Initially this will create more CI splits, but once the CI has been split into a free CI in the same CA, the original and new CIs will have more freespace available. Thus, split activity will tend to stabilise, and CA splits can be reduced without having to sacrifice more disk space.

Action Required:
The file should be re-defined and re-loaded. If inserts are likely to continue, you should increase FRSP CI and/or FRSP CA first.

07. CI SPLITS EXIST
SEV=1 only:
Applies to KSDS/AIX files only. The file has experienced CI splits in the Data and/or Index component, however the number of splits has not exceeded the CI SPLIT warning threshold defined by the CBLNAME field CBLVCSCI.

A small number of CI splits is not necessarily a problem since it may indicate that the amount of CI freespace is not excessive. In addition, when a CI is split, the result is two CIs which are ready to receive additional records. The amount of additional free space caused by the original split will usually accommodate more records than the CI was able to accommodate after initial loading of the file.

Total elimination of CI splits (by allowing additional freespace) may adversely affect sequential processing. This is because the additional freespace in each CI will result in fewer data records per CI and consequently fewer records transferred with one physical I/O operation.

Action Required:
None.

08. CI SPLITS TOO HIGH (nn PC OF INSERTS)
SEV=2/3 only:
Applies to KSDS/AIX files only. If either of these messages are displayed, then the percentage of inserts causing CI splits exceeds the CBLVCSCI threshold (Note that when the number of inserts is less than 100, the actual number of CI splits is compared with the CBLVCSCI threshold).

◊ SEV=3. The number of CI splits per 100 inserts is greater than the threshold (and also greater than 30). This indicates that there is insufficient reserved freespace or that the file is overdue for re-organisation.

◊ SEV=2. The number of CI splits per 100 inserts is greater than the threshold but less than 30 (Note that if the chosen value for CBLVCSCI is greater than 30 then the SEV 2-08 message is suppressed).

CI splits are not always avoidable on files with high insert activity, however, an excessive number of CI splits causes degradation at split time, and also at retrieval time. A high number of CI splits is usually a prelude to the occurrence of CA splits.

Possible reasons for this message are as follows:

◊ The CI size is incorrect for this file.
◊ The CI freespace value is incorrect or inadequate for the amount of insert activity.
◊ The file is not being backed up and re-organised frequently enough.

Action Required:
The file should be re-defined and re-loaded. If inserts are likely to continue, you should increase FRSP CI and/or FRSP CA first.

09. FILE GETTING/IS FULL
SEV=1 only:
Applies to KSDS/AIX files only. The current space occupied by the file has reached the CBLVCPCF threshold (default 85%).

This message may be cause for concern if further record insertions are likely to occur.

The PCNT field (column) will also show "**nn.n**", "** ALL**" or "**FULL**" (If the file is full and no secondary allocation is defined then "**FULL**" is displayed. However, when secondary allocation is defined "** ALL**" is displayed).

If the file is cyclical in growth in that it starts off small, accumulates records for a period of time and is then emptied, it may not be feasible to allocate enough primary space to hold the maximum number of records the file could ever possibly contain. Year to date, or transaction files tend to experience this type of growth. For this type of file, tolerable primary and secondary allocation values need to be found which will not cause excessive allocation degradation. At the same time, they must not tie up an excessive amount of yet unused space.
**Action Required:**

None, except if additions to the file are expected and there is no secondary allocation. In this case, re-define and re-load the file with new primary and/or secondary allocation values.

### 10. IMBED COSTS nn PC OF DATA SPACE

**SEV=1/2/3:**

- **SEV=3:** There is more than one small data CA in use and IMBED uses at least 25 percent of the data area.
- **SEV=2:** There is more than one small data CA in use and IMBED uses between 10 and 25 percent of the data area.
- **SEV=1:** There is more than one small data CA in use and IMBED uses less than 10 percent of the data area.

See **IMBED** and **REPLICATE** in Additional VSAM Information for a description of IMBED.

**Action Required:**

For small files (less than one cylinder), remove IMBED. For larger files, increase the primary and/or secondary allocation size, in order to maximise CA size.

### 11. INDEX CISIZE IS EXCESSIVE

**SEV=2 only:**

Generally, an overlarge index CI size is not a very serious condition, as the index component is relatively small.

It is essential that index CI size is not too low, otherwise whole data control intervals are unusable.

See **LEVELS OF INDEX** in Additional VSAM Information.

**Action Required:**

None immediately. When next defined, either let VSAM default the index CI size, or supply the lower one recommended by CBLVCAT.

### 12. INDEX EXCPS DWARF DATA

**SEV=3 only:**

The number of index EXCPS stored in the catalog is at least twice that of the data EXCPS.

Performance has already suffered greatly (if the SEV 2-04 message is also displayed, the degradation will continue).

The condition is caused by insufficient buffer space for the index. Increasing the BUFSP value is the usual cure, however, CBLVCAT may recommend changes to the data and/or index CI sizes in preference.

See **BUFFER SPACE** and **INDEX CISIZE** in Additional VSAM Information.

**Action Required:**

Alter the defined BUFSP, or increase the BUFSP or BUFNI parameter on your JCL overrides.

### 13. INDEX EXCPS EXCEED DATA

**SEV=2 only:**

The number of index EXCPs is at least equal to, but not double, the number of Data EXCPs (in the case of it being double, SEV 3-12 would be displayed).

Although the condition is likely to be less serious than for SEV 3-12, it is possible that recent additions to the file may have increased the number of index levels, thus making the current BUFSP inadequate. If such a change has occurred recently, the overall comparison of index and data EXCPS does not reflect the current situation.

**Action Required:**

Alter the defined BUFSP, or increase the BUFSP or BUFNI parameter on your JCL overrides.

### 14. KSDS/AIX/VRDS HAS NO IMBEDDED FREE SPACE

**SEV=1 only:**

For KSDS and VRDS files, no space has been reserved for record insertions.

For AIX files with the UPGRADE attribute, no space has been reserved for inserts caused by additions to the base cluster (AIX files without UPGRADE do not require freespace).

**Action Required:**

None, if no inserts are expected.

If additions are to be made, the file should be re-defined and re-loaded. You should decide the filesize percentage increase required between re-organisations, and run a LISTVCAT TUNE with an appropriate GROWTH value.

### 15. nn CYLS CAN BE RECOVERED WHEN TUNED

**SEV=1/2 only:**

- **SEV=2:** 50 percent of the current total allocation, or at least 10 cylinders (or MAX-CAs), may be recovered.
The SEV 2-15 condition is not satisfied, but 25 percent of the current total allocation, or at least 2 cylinders (or MAX-CAs), may be recovered.

**Action Required:**
None necessary.
If you want to recover space, the file must be re-defined with the recommended changes.

### 16. NRECS IS ESTIMATED

#### SEV=1 only:
The catalog statistics are invalid for this file.

This generally occurs when the file has been accessed by physical control interval, instead of logically by record.

CBLVCAT estimates the number of records from the space in use and the defined average record length (see Guide to VSAM Tuning for details).

**Action Required:**
None. For information only.

### 17. SEC ALLOC DEFINED AS ZERO

#### SEV=1 only:
No secondary allocation or expansion is possible for this file.

This may be valid, as static files require no secondary allocation or some products may not allow it for their files.

**Action Required:**
None. For information only.

### 18. SEC EXTENTS EXIST

#### SEV=1 only:
Secondary extents have been acquired, however, the warning threshold (option CBLVCALW) has not yet been reached.

**Action Required:**
None. For information only.

### 19. SEC EXTENTS TOO HIGH

#### SEV=2/3 only:
An unacceptable number of secondary extents have been acquired.

- **SEV=3.** The number of secondary extents has reached the CBLVCALE error threshold.
- **SEV=2.** The number of secondary extents has reached the CBLVCALW warning threshold but not the CBLVCALE value.

When a file's primary allocation space becomes full, VSAM allows the file to be extended if it was defined with a secondary allocation. While this facility is beneficial in preventing program failures due to file full conditions, it can be very costly in performance if it occurs too often.

**Action Required:**
Re-define and re-load the file with a larger primary allocation.

### 20. SHR 4 IS EXPENSIVE

#### SEV=1 only:
Share option 4 should only be used when absolutely essential as it uses the 'track hold' facility and is expensive in terms of machine resources.

**Action Required:**
If multiple write capability is not essential, change to a different share option.

### 21. SPANNED RECORDS

#### SEV=1 only:
Applies only to KSDS/ESDS files.

The use of spanned records should only be necessary for files with the occasional very long record, thus allowing a smaller CI size. In most other cases, the maximum record length and CI size may be changed to allow its removal.

See SPANNED in Tuning Principles for information on the SPANNED attribute.

**Action Required:**
None, for information only (unless the SEV 3-32 is displayed as well, in which case you should re-define with a larger data CISIZE and remove the SPANNED parameter).

### 22. SPEED NOT DEFINED - RECOVERY IS DEFAULT

#### SEV=1 only:
The file was defined with RECOVERY, (or no SPEED define parameter was supplied and the default is RECOVERY).

The RECOVERY overhead is costly and should be avoided unless absolutely necessary.

See SPEED v RECOVERY in Tuning Principles for information on the SPEED/RECOVERY attribute.
Action Required:
None, for information only (unless the Replace RECOVERY with SPEED (or add SPEED to the DEFINE parameters).

23. TIMESTAMP CONFLICT
SEV=3 only:
Only applies to KSDS/AIX files. CBLVCAT has detected a discrepancy in the timestamps of the index and data components.

This could indicate that the data or index has been updated independently (or a system failure not fully completing an update).

Action Required:
Investigation is essential, since there is a strong possibility that the file is now corrupt. The common recovery path is via backup, however, it is possible to retrieve your data by treating the data component as an ESDS file, copying it to a sequential file, sorting it and, finally, loading it into a new KSDS.

24. WRITECHECK IS EXPENSIVE
SEV=1 only:
The WRITECHECK or WCK define parameter is active and has large system overheads (WRITECHECK is unnecessary for today's reliable disks).

See WRITECHECK in Tuning Principles for information on the WRITECHECK attribute.

Action Required:
Remove the WRITECHECK attribute via ALTER and also remove it from your DEFINE parameters, ready for the next file re-organisation.

25. INEFFICIENT DATA CISIZE
SEV=2 only:
There is a poor match between the average record length and the CI size and a CI size is available that will increase file capacity by 25% (For large files, increases of 6 percent are also flagged).

This mis-match wastes disk space and can also affect sequential performance as more I/O will be required.

Action Required:
No immediate action is required.

26. nn CYLS OVER-ALLOCATED
SEV=1/2 only:
Although the file is well-tuned, space is allocated but unused and may be recovered for use by other files.

◊ SEV=2. 50 percent of the current total allocation, or at least 10 cylinders (or MAX-CAs), may be recovered.

◊ SEV=1. The SEV 2-26 condition is not satisfied, but 25 percent of the current total allocation, or at least 2 cylinders (or MAX-CAs), may be recovered.

Action Required:
None necessary.
If you want to recover space, however, the file must be re-defined with the primary and secondary allocations reduced, as appropriate.

27. TUNING FOR RECS/AVRECL CHANGE REQ
SEV=2 only:
Tuning has been influenced by the RECORDS=nnn/ AVLRECL=nnn tuning parameters.

Action Required:
None. For information only.

28. TUNING FOR DEVICE CHANGE REQUEST
SEV=2 only:
Tuning is for a different DASD device, in accordance with the supplied DEV=nnnn parameter.

Action Required:
None. For information only.

29. TUNING FOR GROWTH REQUEST
SEV=2 only:
Tuning has been influenced by the supplied GROWTH=nn or FRSPCA=nn / or FRSPCI=nn tuning parameters.

Action Required:
None. For information only.

30. TUNING FOR CISIZE CHANGE REQUEST
SEV=2 only:
Tuning has been influenced by the supplied CISIZE or CIMIN / CIMAX tuning parameters.

Action Required:
None. For information only.

31. AVLRECL: DEFINED=nnnn ESTIMATED=nnnn
SEV=2 only:
CBLVCAT calculated the range of possible average record lengths based on the space used and number of records. It has found that the defined length does not fall within this range.

The message is only displayed if the estimated average record length is +/- 50% of the defined average record length.

Action Required:
None is essential.
If you know that the actual average length is lower than CBLVCAT’s estimate, changing the defined average record length via \texttt{ALTER}, or specifying \texttt{AVLRECL=nnn} can improve CBLVCAT’s tuning recommendations.

32. FILE SHOULD NOT BE SPANNED
SEV=3 only:
Performance will be improved if the CI size is increased and the \texttt{SPANNED} attribute removed.

Action Required:
Re-define and re-load the file using CBLVCAT’s tuning recommendations.

---

Warning Messages

The format of warning messages is:

e.g.

\[
\text{*** WARN nnn *** text of message}
\]

The following messages are displayed where appropriate (Return code 06 is also given but the operation will continue processing).

01. \texttt{nn = RETURN CODE FROM CBLVCAT}
See the \texttt{Return Code section} of this chapter for an explanation of \texttt{nn} (The listing may contain other warning or error messages).

02. \texttt{INVALID RBA FOUND IN CATALOG}
A return code of X’08’ with error code X’20’ has been encountered during Request Macro processing (CBLVCAT continues processing but users are advised to investigate the Catalog corruption and act accordingly).

03. \texttt{VOLUME GETTING/IS FULL}
A return code of X’08’ with error code X’20’ has been encountered during \texttt{LISTVTOC} with \texttt{FREETAB} operation displays the \texttt{Percentage tracks used}. This value has reached the \texttt{CBLVCPCV} threshold.

04. \texttt{TUNING CANCELLED - NO VALID CISIZE}
A \texttt{CISIZE} or \texttt{CIMAX} restriction parameter has been supplied, and it is impossible to choose a CI size without converting the file to \texttt{SPANNED}.

05. \texttt{volser - DYNALLOC OPEN ERROR nnn}
For ICF catalogs only. Dynamic allocation of the VVDS of one of the associated volumes has failed (See the \texttt{MOUNT / NOMOUNT} options).

06. \texttt{volser - VVDS NOT OPEN}
For ICF catalogs only. An associated VVDS has been dynamically allocated, but the open has failed (The pack may have been dismounted).

07. \texttt{volser - NO ENTRY IN VVDS FOR THIS FILE}
For ICF catalogs only. An associated VVDS has been dynamically allocated and opened, but there is no entry for the file (The VVDS is probably out of step with the ICF Catalog).

08. \texttt{DATA NOT AVAILABLE}
Normally, this error is caused by another program having exclusive control of the Catalog. However, it can also occur when CBLVCAT issues a read request to VSAM and does not receive expected type of record (In this case, a count of such logic errors is displayed in the catalog summary and also at the end of the report as a \texttt{WARN 009} message).

09. \texttt{nn READ ERRORS - LOGIC ERROR}
A count of \texttt{WARN 008} message). messages for the catalog. This may be caused by corruption of the catalog.

10. \texttt{NO FILES SELECTED}
Check \texttt{SUBSET} parameter(s) for undesired specification(s).

11. \texttt{NO FILES TUNED}
A \texttt{TUNE} run has resulted in none of the selected files requiring tuning at the specified \texttt{SEV} level (Unless the \texttt{SEV} parameter is specified, \texttt{TUNE} effects an implied \texttt{SEV=2}).

12. \texttt{DATA SPACE GETTING/IS FULL}
This message applies to \texttt{non-ICF} catalogs only. The \texttt{dataspace(s)} defined to the VSAM catalog is(are) full or nearly full (The \texttt{PCNT} field in the volume summary indicates the percentage of defined space that is currently used by files). The dataspace is regarded as nearly full once the \texttt{PCNT} value reaches the \texttt{CBLVCPCV} threshold (default 85% - See \texttt{CBLNAME}).
This message may be a cause for concern if files defined to the catalog are likely to grow. (Additional dataspace, if available, can be defined to the Catalog).

13. **nn READ ERRORS - TOTAL LOGIC ERRORS**
A count of WARN 008 messages for all catalogs read see WARN 008 and WARN 009 above). This may be caused by corruption of the catalog (as this is a potentially serious warning, Return Code 14 is set).

14. **DEFAULT HIGH LMAX IS DEFINED**
A default maximum record length has been detected. CBLVCAT assumes that the defined maximum record length is correct, as to change it requires local knowledge (If the maximum record length is incorrect it should be corrected before attempting to tune, as it will have an adverse effect on tuning).

15. **LARGE AVLRECL CHANGE**
The CBLVCAT estimated average record length is at least +/-50% of the defined average record length. It is recommended that the actual value is obtained (Users of SELCOPY can contact CBL for a job to calculate an accurate average record length). See AVLRECL Estimation in Tuning Considerations for information on average record length.

16. **LARGE ALLOC CHANGE**
The CBLVCAT estimated allocation is at least +/-50% of the current total allocation. Local knowledge, of previous and intended use of the file, is required before making large changes to allocations (This warning may be accompanied by WARN 015 and/or WARN 017. Attention to these warnings could address the allocation issue).

17. **LARGE NRECS CHANGE**
The CBLVCAT estimated number of records is at least +/-50% of the current number of records. Tuning for reusable files, involves an estimation of the maximum number of records possible considering the allocation (It is recommended that the user checks the estimated number of records). See Guide to VSAM Tuning for further details.

18. **TUNING REQUEST FOR MULTIVOLUME**
A request has been made to tune a multivolume file and CBLVCAT does not provide tune information for multivolume files. Volume allocation is dependent upon local management standards as well as system programming requirements. Therefore, it requires user input to decide on the best volumes to use. (A LISTVTOC report will however give the user free space information on the relevant volumes which can be helpful in deciding file allocations).

19. **SKELETON DECK ONLY - ATTENTION REQUIRED**
A reorganisation deck has been requested and the resulting skeleton jobstream requires user attention before submission (Removal of the warning message is then taken as acceptance of responsibility for the job by the user, who should ensure that adequate back-up exists for the file(s) being reorganised).

20. **MOD DSN/NEWDSN ARE EQUAL**
The DSN and NEWDSN arguments, for a MOD operation, are equal and therefore the MOD operation is ignored.

21. **macro MACRO ERROR RC nn (HEX) - cuu**
CBLVCAT has received an unexpected return code from one of its internal assembler macros. The processing continues as other non-dependent operations follow.

22. **CONFLICTING CATALOG STATISTICS**
General warning returned by CBLVCAT when unexpected values are found in the VSAM/ICF catalog or VVDS. Should not occur - contact CBL.

Error Messages

Control card **syntax errors** are detected before any CATALOG or VTOC processing commences. All control cards must be syntactically correct before any commands are executed (unless **FAIL=IGNORE** is used to force CBLVCAT to run all syntactically correct commands). Some errors, however, will always cause CBLVCAT to cancel as it may be impossible to attempt any further operations. The format of error messages is as follows:

```
  e.g.

  *** ERROR nnn *** text of message
```

The following errors all terminate the current operation and result in a return code 52 (The action taken depends on the **FAIL** parameter, which defaults to **FAIL=CANCEL**).

01. **DEVnnn No lub found (VSE systems only)**
    **DEV=nnnn** has been specified for **DEL, LISTVTOC** or **MOD** and there is no program logical unit (SYS000 - SYS254) currently assigned to that particular drive.

02. **SYSnnn UA or IGN (VSE systems only)**
    **SYS=nnnn** has been specified for **DEL, LISTVTOC** or **MOD** and is **unassigned** or assigned to **ignore**.

03. **SYSnnn NOT DASD - cuu (VSE systems only)**
    **SYS=nnnn** has been specified for **DEL, LISTVTOC** or **MOD** and is assigned to ‘**cuu**’, which is not a disk drive.

04. **VTOC Open Error**
The **CVH** (Common VTOC Handler) routine failed to open the VTOC.

05. **ERROR READING VOL LABEL**
The LISTVTOC, MOD or DEL operation cannot read the DASD Volume Label.

06. ERROR READING F4 LABEL
The LISTVTOC, MOD or DEL operation cannot read the first record within the VTOC (This usually occurs on new, uninitialised volumes).

07. ERROR READING F1 LABEL
The LISTVTOC, operation has encountered an I/O error reading the VTOC.

08. INVALID QUERY
An unsupported parameter has been supplied to the QUERY operation (only CBLNAME is valid).

09. INVALID REPORT
The first parameter supplied to the REPORT operation refers to the type of report that you are defining (Valid parameters are: VCAT or VTOC).

10. LIST OPERATION ONLY FOR ALL (VSE systems only)
DEV=ALL or SYS=ALL has been specified for a DEL or MOD operation, and support is for the LISTVTOC, operation only.

11. INVALID FAIL PARAMETER
The FAIL parameter has been supplied with an unrecognised argument (Valid arguments are: CANCEL, EOJ, IGNORE and IGN).

12. REPORT EXCEEDS PAGEWIDTH
Pagewidth (ie. Report width as opposed to parameter PAGEWIDTH) in this release, is equivalent to 250 (Change your report parameters to display fewer fields, or reduce the spacing between them).

13. CBLV PROGRAM NOT AUTHORISED (MVS systems only)
ERROR 13 is returned if the CBLV program is not authorised (Link Edited with option AC=1) or is not Link Edited in an authorised load library, and a catalog listing is requested. In releases of CBLV prior to 2.10, this would cause an abend IEC1611 followed by ERROR 051.

14. INVALID DEV PARAMETER (VSE systems only)
The LISTVTOC, DEL or MOD operation has a DEV parameter with an invalid argument (The argument format must be cuu or X'cuu').

15. INVALID OPERATION - xxxxx
See the SUMMARY of Syntax for the list of CBLVCAT’s operation words.

16. INVALID PARAM - xxxxx
This parameter is unrecognised by this release of CBLVCAT.

17. unused

18. VOL REQD FOR ALLFILES (VSE systems only)
ALLFILES=YES has been supplied for a DEL operation (As this clears all the files from the volume, the VOL parameter is mandatory as a safety measure).

19. DSN OR ALLFILES REQD ON DEL CARD (VSE systems only)
The DEL operation has no parameter supplied to select the file or files to be deleted.

20. NO FORMAT 1 RECORD (VSE systems only)
For the MOD or DEL operation. There is no entry found in the VTOC for the DSN specified.

21. NEWVOL, NEWOWN OR DSN REQD ON MOD CARD (VSE systems only)
The MOD operation has no parameter supplied to select the modification to be performed.

22. NEW FILE ALREADY EXISTS (VSE systems only)
For the MOD operation only. The file specified by the NEWDSN parameter exists already.

23. INVALID EXP - TEMP OR PERM ONLY (VSE systems only)
For the MOD operation only. The EXP parameter has an invalid argument. The only values permitted are EXP=TEMP or EXP=PERM.

24. NO MODIFICATION TO BE PERFORMED (VSE systems only)
On a MOD operation the control statement is incomplete (Check for missing DSN or VOL parameter).

25. VOL AND OWNER CONFLICT (VSE systems only)
On a LISTVTOC, DEL or MOD operation, both VOL and OWN are specified, but the pack mounted only satisfies one of the parameters.

26. unused

27. cuu - NO PUB ENTRY (VSE systems only)
On a LISTVTOC, DEL or MOD operation, DEV=cuu or refers to a non-existent device.

28. SYSnnn ASSGNED TO cuu (VSE systems only)
29. unused

30. unused

31. ERROR WRITING F1 LABEL (VSE systems only)
   On a DEL or MOD operation, an error has occurred attempting to re-write an F1 label.

32. ERROR WRITING VOL LABEL (VSE systems only)
   On a DEL or MOD operation, an error has occurred attempting to re-write the volume label.

33. INVALID OWN PARAMETER (VSE systems only)
   On a DEL, LISTVTOC or MOD operation, the OWN parameter has an invalid argument (The owner field in the volume label contains 1 to 10 alphanumeric characters).

34. INVALID VOL PARAMETER
   On a DEL, LISTVTOC or MOD operation, the VOL parameter has an invalid argument (The argument must contain six alphanumeric characters).

35. INVALID NEWOWN PARAMETER (VSE systems only)
   On a MOD operation, the NEWOWN parameter has an invalid argument (The owner field in the volume label contains 1 to 10 alphanumeric characters).

36. INVALID NEWVOL PARAMETER (VSE systems only)
   On a MOD operation, the NEWVOL parameter has an invalid argument (The argument must contain six alphanumeric characters).

37. MISSING DDNAME PARAMETER (MVS systems only)
   DDNAME is required for LISTVTOC operations, in order to reference the required volume, unless the VOL or DEV parameter is supplied instead.

38. CARD AREA OVERFLOW
   Data is used from card columns 1 - 71, and column 72 must be blank. Failure to observe this rule is the normal reason for this error (It could occur also if CBLVCAT’s work buffer cannot hold all the parameters for the current operation, however, this is very unlikely, as “white space” is ignored and the buffer can hold 1024 characters).

39. cuu - NO ASSIGNMENTS (VSE systems only)
   On a DEL, LISTVTOC or MOD operation, DEV=cuu has been supplied and there are no SYSnnn assignments to that particular drive.

40. unused

41. INVALID OP ON FBA DEVICE (VSE systems only)
   Modification of the volume label is not supported for FBA disks.

42. INSUFFICIENT STORAGE
   Insufficient virtual storage available (On VSE systems, users should check that the EXEC card has a SIZE parameter that allows sufficient GETVIS space for VSAM).

43. unused

44. INVALID SYS NUMBER (VSE systems only)
   On a DEL, LISTVTOC or MOD operation, the SYS parameter value exceeds the number of Logical Unit Blocks available.

45. INVALID NUMERIC ARGUMENT FOR xxxxx
   The number argument is outside the allowable range for this particular parameter.

46. SORT FAILED
   On a LISTVTOC operation, sorting the extent entries has failed (This error should never occur, so please contact the CBLVCAT query desk at CBL).

47. MAXLRECL LOWER THAN AVLRECL
   On a LISTVCAT TUNE operation, MAXLRECL and AVLRECL parameters have been supplied, but it is illegal for the average length to exceed the maximum length.

48. ddname - NO DD CARD
   A LISTVTOC or LISTVCAT operation has been invoked via the DDNAME parameter and the corresponding DD statement has not been supplied in the JCL.

49. unused

50. CATALOG NOT VSAM OR ICF
On a LISTVCAT operation, a VSAM KSDDS file has been opened successfully, but CBLVCAT does not recognise it as a catalog (Check your job control statements against the DDNAME parameter).

51. VSAM nnn (DEC) OPEN ERROR
   A LISTVCAT operation has failed with a VSAM open error (See the appropriate VSAM Messages and Codes manual). The text for the VSAM return code will generally give sufficient details to correct the error.

52. VSAM nnn (DEC) POINT ERROR
   For VSAM (not ICF) catalogs only. A LISTVCAT operation has failed with a VSAM point error (This error should never occur, so please contact the CBLVCAT query desk at CBL).

53. VSAM RC nnn EC nnn (DEC) READ ERROR
   A LISTVCAT operation has failed with a VSAM read error (See the appropriate VSAM Messages and Codes manual). The text for the VSAM return code and error code combination will generally give sufficient details to correct the error (If this message appears without a return code or error code then please contact CBL).

54. VSAM LOCATE HAS FAILED
   LISTVCAT catalog reference was by REF and VSAM was unable to locate the catalog containing this data set.

55. VSAM DYNALLOC ERROR nnnn-nnnn (DEC)
   LISTVCAT catalog reference was by REF. CBLVCAT successfully located its catalog, but was unable to dynamically allocate it (nnnn-nnnn are the respective Return and Reason codes in decimal).

56. VTOC DYNALLOC ERROR nnnn-nnnn (DEC)
   LISTVTOC reference was by VOL or DEV and dynamic allocation has failed (nnnn-nnnn are the respective Return and Reason codes in decimal).

57. UNIT VERIFICATION ERROR nnn (DEC)
   LISTVTOC reference was by DEV. Unit Verification has failed with a Return Code of nnn (decimal). Please correct the Unit-Name supplied in the LISTVTOC operation.

58. INVALID UNIT NAME PARAMETER
   A LISTVCAT operation has an invalid argument (The argument can contain up to 8 alphanumeric characters).

59. INVALID SECOND VOLN PARAMETER
   Only one VOLn parameter is allowed in a REPORT VCAT operation for a GDG/NONVSAM entry.

60. SYSPUNCH OPEN ERROR
   The DEFINE option outputs to SYSPUNCH, check the status, correct and re-run.

61. INVALID PAGE GEOMETRY
   The geometry specified is not within the defined limits.
   The minimum PAGEDEPTH is 10 lines. The maximum is 32767.
   The minimum PAGEWIDTH is 72 bytes. The maximum is 160 bytes.

62. VVDS FORMAT NOT RECOGNISED
   On a LISTVCAT VVDS=volser operation, a VVDS has been opened successfully, but CBLVCAT does not recognise the format of the record returned (This could occur because the VVDS is corrupt or the format of the VVDS has changed).
   Please contact the CBLVCAT query desk for advice.

63. CONFLICTING CATALOG STATISTICS
   This error should not occur, please contact the CBLVCAT query desk.

64. macro MACRO ERROR RC nn (HEX) - cuu
   CBLVCAT has received an unexpected return code from an internal assembler macro.

---

### Expiry Warning

When the CPU date is within four weeks of the product expiry date, an expiry warning message will be reported on the operator console, for every execution of CBLVCAT. Thus, to most installations, it will be well worth having the up-to-date release in place before the warning commences. The format of the warning is as follows:

```
e.g.
CBLVCAT REL N.NN     *** WARNING ***     EXPIRY DATE --- N JUN YYYY
```

This warning is not given on the SYSLST/SYSPRINT file, although the product expiry date is always reported at the bottom of SYSLST/SYSPRINT file.
Error Messages

When CBLVCAT detects a control card or run time error the appropriate message is logged to the operator console (in addition to the message reported in the listings). This can be suppressed by setting CBLNAME switch CBLVCSW4 to X'03' or using the run-time option.

Wrong Volume

For VSE systems only. The following message is displayed when the drive contains the wrong volume:

*** WRONG VOL=volume1 SYS=nnn volume2 CUU 'owner' ***

Drive CUU currently holds volume 'volume2' which has an ownerid of 'owner' and was referenced via SYS=nnn. The drive should hold 'volume1'. The operator may reply:

<table>
<thead>
<tr>
<th>Option</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;enter&gt;</td>
<td>the current FAIL option is used.</td>
</tr>
<tr>
<td>NEWPAC</td>
<td>CBLVCAT retries the operation and assumes the correct volume has been mounted and made operational.</td>
</tr>
<tr>
<td>IGNORE</td>
<td>the operation is ignored and FAIL=IGN is now current.</td>
</tr>
<tr>
<td>EOJ</td>
<td>terminate the job immediately.</td>
</tr>
<tr>
<td>CANCEL</td>
<td>cancel the job.</td>
</tr>
</tbody>
</table>

Miscellaneous Messages

*** nnn SEC EXTENTS ***
The number of secondary extents acquired has reached the CBLVCALE error threshold. The SEV 3-19 message will always accompany this warning.

** nnn CI SPLITS**
The number of CI splits has reached the CBLVCSCI threshold.

** nnn CA SPLITS**
The number of CA splits has reached the CBLVCSCA threshold.

VOLn=volser
The file referred to resides on more than one volume, or a volume other than the catalog volume.

CANDIDATE VOLn=volser
The file referred to is eligible to be expanded onto this volume, but has not grown sufficiently to do so.

EXTENT nnn nnn/nn - nnn/nn
Extent information as selected by the CBLVCEXT setting.

ASSOC -- AIX=dataset.name.aix

ASSOC -- PATH=dataset.name.path
Associations, other than data or index, printed under DSN on the next line. Printing association details can be enabled/disabled using the ASSOC/NOASSOC option, or by changing CBLVCSW1 in your CBLNAME program.

NOALLOC=nn
VSE systems only. Indicates how much space would be required if all WORK files were opened at once.

MAXLVL=nn
The maximum number of generations for a GDG.

G=NONE
A GDG has no active generations.

Vnn=nnn
The version number and generation number of a GDG.

Return Codes

When CBLVCAT detects an error condition, while running under any operating system, a return code is set and reported in the following format:

*** WARN nnn *** nn = RETURN CODE FROM CBLVCAT

A Minimum Return Code may be defined in the field CBLVCRCM in CBLNAME. Any Return Code that is below this minimum is automatically suppressed and replaced with zero, but is still reported in the SYSLST/SYSPRT file.
e.g.

CBLVCRCM=X'10'  * Suppress Return Codes 1 to 15.

Under operating systems where the Return Code can be tested, e.g. OS, MVS, VSE/SP 2.1 or higher and CMS (with DOS on or off), CBLVCAT will terminate normally, allowing the user to process the Return Code within the JCL or equivalent.

Under operating systems where the Return Code cannot be tested, e.g. DOS, DOS/VS and VSE/SP 1, if the Return Code is less than or equal to 16, then termination is normal, with no indication of a problem except for the *** WARN 001 ***, etc. on the summary.

If the Return Code is greater than 16, CBLVCAT will terminate with a cancel macro, which will flush subsequent steps from the reader to the next // JOB card.

Users of OEM software that does not recognise operating system Return Codes can force cancel, for VSE/SP2 and higher, by setting the CBLVCSW4 X'04' bit on.

CBLVCAT generated Return Codes are as follows:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>00.</td>
<td>Clean Run</td>
<td>No abnormal condition encountered.</td>
</tr>
<tr>
<td>04.</td>
<td>No Files Selected.</td>
<td>The subsetting criteria used has caused no files to be selected.</td>
</tr>
<tr>
<td>06.</td>
<td>Warning Message Occurred.</td>
<td>The job has run to completion but has generated a warning message.</td>
</tr>
<tr>
<td>08.</td>
<td>Sev1 Message Occurred.</td>
<td>There is a Severity 1 message in the output.</td>
</tr>
<tr>
<td>10.</td>
<td>Sev2 Message Occurred.</td>
<td>There is a Severity 2 message in the output.</td>
</tr>
<tr>
<td>12.</td>
<td>Sev3 Message Occurred.</td>
<td>There is a Severity 3 message in the output.</td>
</tr>
<tr>
<td>14.</td>
<td>Severe Warning.</td>
<td>The job has run to completion but has detected a condition which should be given immediate attention by the user.</td>
</tr>
<tr>
<td>52.</td>
<td>Error Message.</td>
<td>A run-time or control-card error has occurred, and the job terminated. The cause of the problem will have already been detailed earlier in the SYSLST/SYSPRINT output.</td>
</tr>
</tbody>
</table>
Technical Information

This chapter contains a brief overview of CBLVCAT, installation, discussions on tuning principles, and more detailed explanation of various VSAM topics.

Installation

1. Introduction
2. The Distribution Material
3. The Operating Environment
4. System Components
5. System Considerations
6. Year 2000 Compliance

Introduction

This section provides an overview of the installation of CBLVCAT and is not designed to be used as the only source of reference for product installation (The separate installation guide provided within the distribution material gives full details of current installation procedure).

The Distribution Material

For each zSeries mainframe platform, there exists a CBL product .ZIP bundle file containing all CBL software products (including CBLVCAT), and machine readable documentation in PDF format (Install instructions and New Features Document.)

The product binary file is a single file containing object modules, installation jobs, HELP files and other miscellaneous product related files.

Once the installation is complete, refer to the Introduction chapter of this manual for an explanation on running CBLVCAT under your particular operating system.

The Operating Environment

Processors Supported
The following processors are suitable:
Any zSeries, System/390, ES/9000, System/370, or equivalent.

Operating System Support
- Any VSE type system - z/VSE, VSE/ESA, VSE and DOS/VS.
- Any MVS type system - z/OS, OS/390, MVS/ESA, MVS/XA, MVS, OS and TSO.
- VM/CMS subject to system considerations.

DASD Support
The following disk types:
0671, 231x, 3310, 3330, 3340, 3350, 3370, 3375, 3380, 3390, 9332, 9335, 9336 and 9345.

Addressability
CBLVCAT supports 31-bit addressing and is link edited as AMODE(31),RMODE(24) on all platforms. Unless required otherwise, CBLVCAT always attempts to utilise buffers in above the 16MB line storage.

System Components

The program code is totally contained within the module CBLV.

The small CBLVCAT, CBLVTOC and CBLVTOCM modules, whose only function was to call CBLV, are no longer provided. Whilst these may still function, they are no longer maintained by CBL and it is recommended that CBLV is used.

Long time users please note that when invocation does function through CBLVCAT/CBLVTOC, the CBLNAME option for SYSIN will also work, although it is recommended that new jobs use CBLV for efficiency.
CBLVCAT Build Level Information

CBLVCAT release and build level is displayed in the report footing. Alternatively, to simply establish the timestamp and build level of CBLVCAT, execute CBLV with no control statements and specify -V as a parameter (i.e. in the PARM field).

The build level information is written to SYSLOG/SYSLST. e.g.

   Build Level=156 2004/01/22 13:18 (Latest change).

System Considerations

VSAM and ICF catalogs are processed in read-only mode. VTOCs are also processed read-only, except for the VSE modification commands (which may be disabled on request).

CBLVCAT is neither re-entrant or re-usable.

MVS type systems
Install in an authorised library and authorise CBLV.

Do not include update the install link edit job to includes any parameters which would result in re-entrant or re-usable load programs.

CBLVCAT’s internal organisation is such that parts of the program, which need to be in storage most of the time, are on a page boundary relative to the start of the module. Thus, a reduction of the amount of paging is possible by using the MVS Link Editor's ORDER statement to request a page boundary for the start of the module. Therefore, the statement ORDER CBLV(P) is included in front of the NAME CBLV(R) record in the CBL products install link edit job.

VSE type systems
Do not run CBLVCAT from the Shared Virtual Area (SVA).

A programmer logical unit number (SYSnnn) needs to be assigned to SYSIPT in the processing partition.

VM/CMS
The CMS/VSAM product is required for VSAM catalog reports.

CBLVCAT must be run from a DOSLIB for VTOC reports.

If CBLV is invoked more than once in the same job, EXECOS CBLV should be used. This enables a fresh copy of the program to be loaded at each execution, which is necessary as CBLV is neither re-entrant or re-usable.

Year 2000 Compliance

Definition of compliancy for CBLVCAT:

Operates correctly for dates either side of the change to the 21st century.

Operates correctly for both “YR2000 Ready” (enabled) levels of the Operating Systems and earlier releases.

See appropriate IBM documentation for further details regarding enabled levels of IBM software. (e.g. G225-4508-13 "Special Issue - VSE and Year 2000" and GC28-1251-05 “The Year 2000 and 2-Digit Dates”.)

Background information:

The following facts, relating to the Operating Systems, have been taken into account when addressing CBLVCAT's YR2000 compliancy:

Dates in ICF and VSAM Catalogs
ICF catalogs since DFP Ver 2.3.0 have included century information for the dataset's Creation and Expiry dates.

VSAM catalogs under MVS will not be supported when the system date is beyond 1999.

VSAM catalogs under VSE continue to hold the creation and expiry dates with 2 digits for year. Year 2000 enabled releases of VSAM interpret the century from these dates using a sliding window technique:

Create Date:
Sliding Window in the range -79 to +20 relative to Current Date.
   e.g. In 1997, the window is 1997-79 to 1997+20 which equals 1918 to 2017. Therefore, a create date with YY=17 is interpreted as 2017 but create date with YY=18 is interpreted as 1918.

Expiry Date:
Sliding Window in the range 0 to +99 relative to Create Date.
   e.g. For 1997/001, window is 1997+0 to 1997+99 which equals 1997 to 2096. Therefore, an expiry date with
YY=17 is interpreted as 2017, with YY=18 is interpreted as 2018. However, an expiry date with YY=98 is interpreted as 1998, with YY=97 is interpreted as 1997 but create date with YY=96 is interpreted as 2096.

Note:
A VSAM definition will fail if the expiration date (TO= parameter) is outside the acceptable range of 0 - 99 years.

Dates in VTOCs
The Year in Creation and Expiry date is held in 1 byte binary, enabling values of 0 to 255 relative to 1900 to be stored.

SAM and VSAM Permanent Files
◊ 99/366 is never considered as expired.
◊ 99/365 when explicitly specified (VSE DLBL or MVS DD EXPDT for SAM, TO= for VSAM) is considered as never expired.
◊ 99/365 when calculated from a retention period is considered expired when earlier than or equal to current date.

CBLVCAT Operations Reflecting 4 Digit Year:

LISTVCAT Reports
◊ Date of execution in Report Heading.
◊ TIMESTMP/DEFINED columns dates.
◊ For VSAM Catalog summary reports, the VSAM Data Space TIMESTMP date.

LISTVTOC Reports
◊ Date of execution in Report Heading.
◊ ACCESSED column dates.
◊ CREATED column dates.
◊ EXPIRES column dates and interpretation of PERMANENT and EXPD file dates.
◊ OLDEST and LATEST dated files in the VTOC in the VTOC summary report.

Subset Options
◊ LISTVCAT & LISTVTOC filter HIDATE=/LODATE= dates.
◊ LISTVTOC filter HIEXP=/LOEXP= dates.

VSE SAM file MODify
◊ EXP=PERM to change the files expiry date to 99/366 (PERMANENT).
◊ EXP=TEMP changes the files expiry date to 01/001 (TEMPORARY).

IDCAMS DEFINE Output
◊ DEFINE CLUSTER expiry parameter TO(....).

Tuning Principles

1. Approach to ALLOCATION
2. Approach to RECORDSIZE
3. Approach to DATA CISIZE
4. Approach to INDEX CISIZE
5. Approach to IMBED/REPLICATE
6. Approach to SPEED v RECOVERY
7. Approach to WRITECHECK
8. Approach to SHAREOPTION
9. Approach to SPANNED
10. Approach to FREESPACE
11. Approach to BUFFERSPACE

Approach to ALLOCATION
CBLVCAT will suggest how much space to allocate based on the current file statistics and your intuitive overrides (see Summary of Syntax for a list).

CBLVCAT’s recommended values take the following into account:

• Enough primary space to accommodate all of the records currently on the file.
• Enough primary space to include growth and free space.
• Enough secondary space to include growth and free space.

Ensuring contiguous allocations is a matter out of CBLVCAT’s control. You may, however, request that CBLVCAT reports file extents so that it can be determined when allocations are scattered and non-contiguous (See CBLVCEXT in CBLNAME and the CBLVCEXT option for run time over-ride).
Where large files are concerned, there are other matters over which CBLVCAT has no control (this is because they are yours to choose). It will always try to suggest a primary allocation quantity, which will include enough space for all of the existing file in its tuned state. If the file is too large to be contained on a single volume, it will present a primary allocation value the size of one volume, indicate the number of primary volumes required and then present the remaining space required in the secondary allocation value. You will be armed with enough local knowledge to decide how the total amount of space recommended by CBLVCAT will actually be distributed (See the CYLMAX option).

CBLVCAT will never suggest allocation by RECORDS. It will always suggest TRACKS or CYLINDERS for CKD disks and BLOCKS for FBA disks. TRACKS (or BLOCKS which are not MAX-CA multiples) are only recommended for files which will occupy considerably less than 1 cylinder and will be such that they result in the best CA size. Incidentally, even VSAM converts your RECORDS specification to tracks, cylinders or blocks and stores the converted value in the catalog.

**Approach to RECORDSIZE**

CBLVCAT will not normally suggest that your maximimum record length should be changed. (However, CBLVCAT will make the maximum record length equal to the average record length if the user supplies an AVLRECL=nn override value greater than the defined maximum record length). This is strictly a choice which is dictated by the application. However, it will try to point out an inaccurate average record length by considering the amount of file space now occupied by records.

In calculating the average record size, keep in mind that there may be unrecorded or inaccurate statistics in the catalog due to files being open, or to previous abnormal program terminations. Files which are badly in need of reorganisation may also have an adverse impact on CBLVCAT's average record length calculation. SELCOPY users could use that product to determine the average length of a loaded file - ring the SELCOPY query desk if assistance is required.

Another area which will negatively affect CBLVCAT's recommendations, is the specification of maximum records sizes which are much greater than the actual records in the file.

**Approach to DATA CISIZE**

As with other file definition attributes, selecting a Control Interval size is a matter of considering trade-offs.

CBLVCAT will attempt to select a data CI size which best utilises the device. However, it does not take into consideration on-line processing unless you specify your intuitive override (see Summary of Syntax for a list).

The recommended values try to take the following into account:

1. The operating system, which affects the physical record sizes.
2. The device geometry (track size, MIN-CA), which determines the most effective CI sizes.
3. Free space for future record insertion. If the file experiences insert activity after it is initially loaded, there should be enough free space to insert at least one record into the CI. If this is not practical due to large record sizes and/or few records per CI, then free space in the CA should compensate.

CBLVCAT will always suggest cylinders (MAX-CA) values for large files, or tracks (MIN-CA) multiples) for very small files. The suggested value will yield the proper CA size.

When calculating a CI size for a file containing records of varying length, CBLVCAT takes the optimistic approach, calculating the number of records per CI based on the value CISIZE-10.

**Approach to INDEX CISIZE**

CBLVCAT may indicate a change to the index CI size of a KSDS. Under most circumstances, IDCAMS will select the same CI size as CBLVCAT, if allowed to do so. There may be situations however, where CBLVCAT and IDCAMS will differ. Here, it is better to adopt CBLVCAT's value.

Unless CBLVCAT sees an index CI size which is too small, the revised value will be due to a change in data component allocation units (from tracks to cylinders or vice-versa). In any case, when an Index CI size change is indicated, it will result in different buffer and disk storage requirements.

See the Additional VSAM Information section.

**Approach to IMBED/REPLICATE**

CBLVCAT only tells you to remove IMBED and REPLICATE if the file is small. The BUFSP value will more than compensate for these attributes.

If the file is large, CBLVCAT will not suggest you remove these attributes. The reason for this is that the catalog statistics probably won't reflect the worst conditions in the file's life (e.g. just prior to reorganisation or at the end of the file's growth cycle). Also, if
these attributes are defined, their presence may be justifiable.

See the Additional VSAM Information section.

**Approach to SPEED v RECOVERY**

The advantages offered by SPEED will always outweigh those offered by RECOVERY. CBLVCAT will always recommend that SPEED be adopted for any file selected for tuning. It is up to you to decide otherwise.

**Approach to WRITECHECK**

CBLVCAT will always recommend that you remove WRITECHECK.

See heading NOWRITECHK in the section TUNE Block of Tuning Output.

**Approach to SHAREOPTION**

CBLVCAT will only warn you that SHAREOPTION 4 is in effect and expensive. It is up to you to evaluate whether a more performance enhancing value can be applied (Even with SHAREOPTION 4, 100% write integrity cannot be assured).

**Approach to SPANNED**

If CBLVCAT finds that a file is defined with the SPANNED attribute and the CI size is greater than the maximum record length (or if it finds that a CI size adjustment can eliminate the need for the SPANNED attribute) it will recommend the attribute is removed.

**Approach to FREESPACE**

Because of the complexity of the matter, there are a number of factors which affect the way CBLVCAT approaches free space. Generally, CBLVCAT will attempt to increase or decrease free space, based on the statistics it finds in the catalog and the percentage of inserts causing splits. In the adjustment attempt, it will use records per CI and CIs per CA in its justification. It will also warn you when the existing FRSP values are incorrect or missing due to arbitrary selections or omissions when the file was first defined.

Any CI free space must allow for at least one record to be inserted. When CBLVCAT sees that the CI is large enough to hold a reasonable number of records, it will recommend a value which will allow enough free space per CI for at least two record insertions. If the records are large, or there are few records per CI, CBLVCAT may suggest zero free space per CI and a corresponding increase in free CIs per CA. No matter what free space values it recommends, you will easily be able to evaluate them (from CBLVCAT’s corresponding comment) in terms of the number of free records per CI and free CIs per CA.

When insert activity fluctuates between tuning runs, you may find that sometimes the recommended values also fluctuate. In these instances, checking the comments in the file’s IDCAMS DEFINE job should give an indication as to how your expectations have been set and how to re-tune to obtain consistent values.

It should be noted that any free space is a general value which, unless extreme measures are taken, applies to the entire file.

See Growth and Freespace section and FREESPACE heading in section Tuning Output for a description of how to influence freespace recommendations.

**Approach to BUFFERSPACE**

The best performance is achieved when JCL is used to specify the most suitable number and combination of buffers, whereas, the worst performance is suffered when the IDCAMS minimum default is used. In view of this, CBLVCAT suggests a compromise buffer space value for file definition. This generalised value will apply to those jobs whose JCL does not take advantage of tailoring the type (data and index) and number of buffers.

When the file is a KSDS, this compromise will yield respectable performance for random access and reasonable (but not super) performance for sequential access. Note that, when specifying a general quantity of buffer space (i.e. BUFSP as opposed to BUFND and BUFNI), VSAM allocates the available buffer space to suit either random or sequential processing. By correctly adjusting the type (index or data) and number of buffers, either by way of JCL, or in the CICS FCT, performance will always be improved (except when SHR(4) is in effect).
**BUFSP for IDCAMS DEFINE** (random processing)
CBLVCAT will recommend a value which will (justifiably) yield good performance for random processing of KSDS and AIX files. The value will result in VSAM allocating two data buffers (the requirement for normal random processing) and at least one buffer for each level of index plus 1. This value, which should be specified in the **BUFFERSPACE** parameter of the IDCAMS DEFINE, is slightly more than one per level of index, leading to the following:

1. Better performance because more of the index set will be in buffers.
2. In case of file growth another level of index can be accommodated.
3. Every I/O requires the operating system to ensure that a certain number of storage pages are fixed in order to receive the transferred data. Storage pages are 2K or 4K in size. CBLVCAT accounts for this and it may suggest a buffer space value which results in better page usage as well as extra index buffers.
4. If the KSDS is processed sequentially and no JCL buffers are specified, VSAM will initially allocate only one index buffer (all that is required). After ensuring two data buffers (the minimum), the remaining allotted buffer space will be used to allocate additional data buffers. After a whole number of data buffers are allocated, any remainder is given back to index buffers (if sufficient).

The resulting mix of buffers could result in more than the minimum 2 data buffers. This may not be enough to activate read-ahead, but could result in a very significant reduction in I/O to the data component.

5. Where KSDS files are opened implicitly, without being specified via JCL, such as when processing via the PATH or using the IDCAMS REPRO naming the data set, reasonable performance can be achieved because the catalog buffer space value recommended by CBLVCAT is better than the IDCAMS default.

6. The KSDS buffer space value recommended by CBLVCAT is correct to achieve respectable performance during direct processing. Once established in the catalog, program or JCL overrides cannot specify less buffer space. In CICS, where files are typically accessed concurrently by more than one user (STRNO is greater than 1), the minimal efficient buffer requirements will be greater than that of the catalog (which is for a single string) and should be attended to in the FCT. If the FCT STRNO value is 1 and insufficient buffers are specified or defaulted, the catalog buffer space will force a better value.

7. VSAM controls the management of LSR pools, not CICS. CICS can only pass the appropriate requests to VSAM. When LSR pools are used, dedicated buffer space is no longer a concern and the value stored in the catalog does not apply.

For ESDS and RRDS files, buffer space for two data buffers is recommended (VSAM's minimum). The justification for this action is covered under the next item.

**BUFSP for JCL override** (sequential processing)
When any file is processed sequentially, the speed at which the file can be processed depends on a number of factors (e.g. CI size, CIs per CA, Number of allocations/extent, Number of CI and CA splits and the Job priority). However the most crucial performance factor is the number of data buffers. Generally, the more data buffers are available, the faster the file is processed.

Re-iterating that the best place to specify buffer space is in JCL, CBLVCAT will suggest JCL values to override the value stored in the catalog. These data buffer values are designed to yield one of three performance levels called **FAST**, **URGENT** and **TURBO**. Where applicable, CBLVCAT will warn you when using the **TURBO** value could impact the rest of your system. In this case, the option should be used with caution.

In summary, where tight control of buffer space is not practiced and controlled in JCL, CBLVCAT will eliminate the guesswork in determining a BUFSP value which is effective.

See also the Additional VSAM Information.

---

**Tuning for IMS/DL1 Databases**

1. IMS/DL1 Data Sets
2. IMS/DL1 Database Processing
3. Database Re-organisation
4. Buffer Pool Allocation

IMS and DL1 are VSAM applications and are subject to the same problems and performance issues encountered by any other VSAM application. These are primarily related to allocation and use of VSAM buffers.

For IMS/DL1, LSR buffer pools should be specified in a file with DDNAME=DFSVSAMP (VSAM Parameters). If DFSVSAMP is omitted, the default values are used which are most likely not appropriate to the VSAM file.

To investigate:

- View VSAM parameters in the DFSVSAMP data set.
- VCAT TUNE listing of all associated VSAM files in the database (may be several).

To fix:
Reorganise VSAM files according to VCAT’s recommendations.
Add buffer information to definition statements in DFSVSAMP.

IMS/DL1 Data Sets

In general, IMS/DL1 databases are comprised of a number of VSAM ESDS and/or VSAM KSDS data sets.

- Databases that have indexes (HIDAM) store index records within KSDS data sets wherein records tend to be quite short.
- The data records of an IMS/DL1 database are generally stored within VSAM ESDS data sets.

Since IMS/DL1 is a VSAM application, processing data from an IMS/DL1 database requires VSAM buffers which, for both batch and on-line CICS processes, are specified via LSR.

For MVS, the LSR buffer pool is referenced by a data set with the DDname DFSVSAMP (VSAM Parameters). Omission of a DFSVSAMP file is not advised as the default buffer allocation is likely to be inappropriate.

Multiple databases may be referenced by any single PSB. Thus, for any IMS/DL1 batch or on-line process, the number, size and type of buffers to be defined must take into account the geometry of each VSAM data set involved to allow maximum efficiency.

IMS/DL1 Database Processing

IMS/DL1 database processing is generally made more efficient by performing the following:

- Re-organisation of the databases so that associated data segments exist in contiguous control intervals.
- Ensure that the buffer pool allocation defined for individual jobs best matches the type of processing to be actioned on the database.
- In extreme cases only, re-organisation of the IMS/DL1 database VSAM data sets.

Database Re-organisation

Where a job's performance has deteriorated over a period of time, the most likely cause is poor organisation of the database’s data records within the VSAM data sets. This can occur as record inserts are made throughout the database.

Even on sequential processing of a database, where related data segments are located on non-contiguous CIs throughout the VSAM file, extra EXCPs are required to load the VSAM buffer pools with CIs for contiguous segments that would otherwise exist in CIs already loaded in the buffer pool.

Similarly, record inserts may eventually cause a CI or CA split in the indexing of a HIDAM database. The effects of this may be that newly created CIs or CAs contain an excess amount of freespace and so more CIs need to be loaded to process the same amount of data segments as before the split occurred.

A database re-organisation will restore order to the data records and so improve processing efficiency.

Buffer Pool Allocation

The following basic principles are true for all efficient VSAM data set processing.

Sequential Processing

For sequential processing, efficiency is improved by allocating more data buffers.

If possible, define a number of buffers to allow VSAM to perform a read ahead of Data CIs. This occurs for VSAM sequential processing where enough free buffers exist that an entire CA may be read into the buffers in a single I/O. i.e.

define a sub-pool of buffers numbering CIs per CA.

e.g. If CISIZE = 4KB and CASIZE = 720KB, then allocate 180 * 4KB buffers.

Therefore, for each database referenced define a sub-pool of buffers numbering CIs per CA. If the buffer sub-pool is shared between database files then even more data buffers would need to be defined.

The CI/CA value is displayed as part of a standard CBLVCAT LISTVCAT report.

Furthermore, CBLVCAT's TUNE function may be used to help define the optimum size and number of LSR buffers for sequential input of an IMS/DL1 database.
CBLVCAT's VSAM data set tuning offers general recommendations based on further record insertions to the data set, current file statistics (number of records, record size, CI and CA splits, etc.) and user supplied overrides.

Most importantly in this discussion, CBLVCAT specifies a choice of JCL overrides for BUFND that may be used to speed up sequential processing. These overrides may equally be applied to buffer sub-pools within the DFSVSAMP data set.

By enforcing restrictions on the CBLVCAT TUNE, the recommended VSAM parameters will be unchanged and the BUFND overrides will reflect the file in its current form. This is achieved by coding the following:

- CISIZE=KEEP
- GROWTH=0
- AVLRECL=KEEP
- MAXLRECL=KEEP

The BUFND overrides are based on the tuned CISIZE and BUFFERSPACE values. Unless these current values are flagged with an asterisk ("*") in the report output, it is unlikely that there is an urgent requirement to re-organise the VSAM files. In particular, re-organising the files with a new CISIZE would require examination and possible update to buffer allocations for other IMS/DL1 jobs on the same database.

Using TUNE option CISIZE=KEEP is recommended so that CBLVCAT's tuning recommendations are based on the existing CISIZE only.

Because we are interested in input only, we do not want CBLVCAT to make tuning recommendations based on further record insertions, therefore GROWTH=0 should be specified.

**Direct Processing (KSDS only)**

Define at least as many Index buffers equal to the number of index levels + 1 for the sequence set. This is necessary to avoid input of the top level index for every direct read which would cause disk thrashing. Specifying more index buffers will increase the likelihood that the required index CIs are already in storage and so reduce the number of EXCPs.

If it is known that consecutive direct input requests are likely to involve records that are located in close proximity to each other, then definition of multiple data buffers may be of benefit. However, in most cases, comparatively few data buffers need to be defined.

Because of the different buffer requirements for each type of processing, it is general good practice to define separate DFSVSAMP data sets containing different buffer pool allocations appropriate to the type of processing and geometry of the VSAM data sets involved.

### Additional VSAM Information

1. MIN-CA and MAX-CA
2. CASIZE
3. CISIZE
4. SECONDARY ALLOCATION
5. IMBED and REPLICATE
6. FREESPACE
7. SPLITs
8. KEY COMPRESSIOn
9. ALTERNATE INDEX and PATH
10. LEVELS OF INDEX
11. BUFFER SPACE
   - BUFSP for Sequential Processing
   - BUFSP for Direct Processing
   - BUFSP for Sequential and Direct
12. SPEED v RECOVERY

### MIN-CA and MAX-CA

These terms, common to both CKD and FBA devices, describe VSAM's use of the track and cylinder concept to optimise performance and to control allocation.

**MIN-CA** replaces the term 'track'.

**MAX-CA** replaces the term 'cylinder'.

MIN-CA and MAX-CA are units of allocation, the size of which depends upon the device being used:

<table>
<thead>
<tr>
<th>DASD</th>
<th>MIN-CA</th>
<th>MAX-CA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BLOCKS</td>
<td>BLOCKS</td>
</tr>
<tr>
<td>0671</td>
<td>63</td>
<td>504</td>
</tr>
<tr>
<td>3310</td>
<td>32</td>
<td>352</td>
</tr>
</tbody>
</table>
* For CKD devices, depends on the size and number of physical records.

**Performance**

Is optimised if there are an integral number of control areas occupying a MAX-CA. CBLVCAT's tuning takes this into account.

---

**CASIZE**

The CONTROL AREA size is selected by AMS from the primary and secondary space values (i.e. TRACKS/CYLINDERS/RECORDS). AMS checks the smaller of the primary and secondary space values against the DASD device's MAX-CA size. If the smaller space quantity is less than or equal to the max-ca, the CA is set equal to the smallest space quantity (rounded up by MIN-CA value with FBA). If the smaller space quantity is greater than the max-ca, the CA size is set to the MAX-CA value.

VSAM acquires space in increments of control areas.

---

**CISIZE**

The CONTROL INTERVAL is VSAM's logical record length.

It holds as many physical records as will fit (non-spanned.) Each CI contains VSAM control information in the last few bytes.

- 4 bytes of CIDF (per CI)
- 3 bytes of RDF (per different record length in the CI).
- 3 bytes of RDF (count of adjacent records of same length).

Therefore, for more than one record per CI, fixed length records have 10 bytes of control information and variable length records have 10+ bytes of control information.
### FIXED LENGTH RECORDS

<table>
<thead>
<tr>
<th>Rec 1</th>
<th>Rec 2</th>
<th>Rec 3</th>
<th>Rec 4</th>
<th>FREE SPACE</th>
<th>RDF</th>
<th>RDF</th>
<th>CIDF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FREE</td>
<td>#</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>___________</td>
<td>___________</td>
<td>___________</td>
<td>___________</td>
<td>__________________</td>
<td>___</td>
<td>___</td>
<td>____</td>
</tr>
</tbody>
</table>

### VARIABLE LENGTH RECORDS

<table>
<thead>
<tr>
<th>Rec 1</th>
<th>Rec 2</th>
<th>Rec 3</th>
<th>Rec 4</th>
<th>Rec 5</th>
<th>FREE SPACE</th>
<th>RDF</th>
<th>RDF</th>
<th>RDF</th>
<th>RDF</th>
<th>CIDF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SPACE</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td></td>
</tr>
</tbody>
</table>
| # = no. of adjacent records of same length.  
* = record length.

A Control Interval (like SAM BLKSIZE) is the unit of transmission between DASD and main storage.

The size of a CI must be a multiple of 512 bytes.

Data CI's include all multiples of 512 up to 8192, then up to 32768 in multiples of 2048.

INDEX CI's may only contain one physical record so the range available depends on the operating system:-

- **DOS/VS**: 512, 1024, 2048 or 4096.
- **DOS/VSE**: all multiples of 512 up to 8192.
- **VSE/SP (or higher)**: all multiples of 512 up to 8192.
- **MVS (prior to DFP 2.2)**: 512, 1024, 2048 or 4096.
- **MVS (DFP 2.2 onwards)**: all multiples of 512 up to 8192 and all multiples of 2048 up to 32768.

AMS will round up for non-multiples of 512/8192.

A Data CI must be large enough to hold a data record of the maximum size specified in the RECORDSIZE parameter (Exception to this being SPANNED records).

**Performance**

For **sequential processing**, large data CI's improve performance by allowing more data to be transferred by one execute channel program (EXCP) operation.

For **direct processing**, large data CI's tend to reduce performance as unnecessary data is transferred to and from primary storage.

Small data CI's tend to cause large index CI's which results in more data CI's per data CA, requiring more index entries in each sequence set index record.

### SECONDARY ALLOCATION

The Secondary allocation, is the amount of additional space to be given to the file when the primary allocation becomes full. The value is specified at file definition time.

The maximum number of **extents** supported by VSAM is 123 (except for reusable non-ICF files where the maximum is 16 per volume). Files in ICF catalogs can have 123 **extents**, regardless of the **REUSE** attribute. If the Dataspaces is fragmented (not enough contiguous space available), the primary allocation can take more than one extent. This will reduce the number of possible secondary allocations. If a file is extended onto a candidate volume a Primary allocation is acquired and not a Secondary allocation.

An allocation (primary/secondary) is carried out on a "best-fit" basis. This means that if the allocation quantity can be located within a single contiguous extent, it will take place in the closest fitting available space. If enough contiguous space is not available, a search is made to see if the allocation amount can be satisfied in more than one extent, but ever more than 5 extents (Please note that some releases of DFP on MVS systems may require contiguous allocation extents when allocation is by tracks. This means that an allocation must be satisfied in a single extent).

**Performance**

Every time VSAM is required to acquire more space for a secondary allocation, it needs to suspend activity on the processing program and find room to expand the file. This can take a considerable amount of time, especially for on-line applications.

A small secondary allocation, for significantly large files that need to expand, will lead to an unrealistic number of secondary extents. This can cause a performance degradation.

If the small secondary allocation is less than a MAX-CA then it will impact performances by resulting in a small CA size.
**IMBED v REPLICATE**

The **IMBED** attribute means that the file is defined to have the lowest level of the index (the sequence set) placed within the data component. It is placed in the first track of each control area for which it exists and is automatically replicated on this track.

![Control Area Diagram]

The **REPLICATE** attribute means that the file is defined to have the index CI's repeated as many times as will fit on a track (within the index component).

**Performance**

- Imbedding the sequence set within the data reduces access arm movement for both sequential or direct processing.
- Index set replication decreases search time when reading index set records directly.

If many index buffers are available, index options have less effect.

**IMBED** is very costly for disk space if the CA size is small. (e.g. if secondary allocation is 2 tracks, then CASIZE is fixed at 2 tracks and IMBED would use 50 per cent of the data area).

**FREESPACE**

For KSDS and AIX only, can be requested at two levels:

**FRSPCI**

A percentage of each control interval can be left free (in bytes). VSAM does **not** round up to a whole number of records.

**FRSPCA**

A percentage of control intervals within each control area can be left free. For VSE the space reserved is rounded up to the next whole number (unless zero is specified), whereas MVS rounds down to the next whole number (or zero).

Freespace is left when the file is loaded and also when it is extended. (i.e. when CI/CA splits occur when in sequential insertion mode).

CI freespace should be as large as the design insertion level. (i.e. estimate the percentage of additions to be made between file reorganisations). Beware of Freespace definitions which give **unusable** areas within a CI. (It is possible to allocate freespace that is not large enough to contain a single record).

Freespace may be altered after the file is loaded. To take full advantage of mass insertion, ALTER Freespace to (0 0) after the load.

An alternate method of handling an uneven pattern of additions, is to define the file as a key-range, leaving room for expansion at the end of each range of keys.

For **FREESPACE(100 100)**, VSAM writes one record to each CI and one CI to a CA.

**Performance**

- For **direct** insertions, make the CI freespace larger than the CA freespace.

  The greater the freespace specification, the more DASD space is required. For **sequential processing**, more I/O operations (with more system overhead) are required to process the same number of records.

  A bad combination of **CISIZ, RECORDSIZE** and **FREESPACE** can cause poor sequential performance if much of the free space is unusable.
Too much freespace could increase the number of index levels, which could increase run time for direct processing.

Too little freespace can cause an excess of (time-consuming) CI/CA splits. After a split, extra time is required for sequential processing because the records are not in physical sequence.

For direct processing, CA splits can increase seek time (Another factor is the additional VSAM overhead required to do the split).

**FREESPACE Recommendations**

CBLVCAT bases its free space recommendations in line with the catalog statistics. If the information is indicative of the file's normal behaviour patterns, then the values it recommends will be correct.

If the information is not indicative of the file's behaviour patterns, then you can influence CBLVCAT to arrive at the correct recommendations through the use of the TUNE sub-parameters. See Tuning Considerations in Guide to VSAM Tuning for details.

The following items will help you to decide when and how to adjust CBLVCAT's free space recommendations:

- When the file being tuned has just been loaded, the catalog will not reflect the amount of insert activity. RECORDS=nnn and GROWTH=nn can be used to adjust the free space required.
- The influencing values should represent the number of records to be loaded, together with the intended growth, due to record inserts, before the next file reorganisation.
- When the file being tuned never has records added to it by way of insert, the catalog will not show insert activity, but CBLVCAT should be assured of this by specifying GROWTH=0.
- These types of files are usually used as table values by your applications. Usually, they are updated as a part of the initial load procedure. In this case, you may also want to use the RECORDS=nnn sub-parameter to influence the allocation recommendation.
- When the file is in use (opened) by another program, such as CICS, the catalog statistics may not be reliable. This is because they will only be updated when the file is closed. Running CBLVCAT when CICS is down, or when the file is closed, will get the correct results. However, the use of RECORDS=nnn and GROWTH=nn will also achieve the desired result.
- When the insert activity to the file is such that inserts are evenly distributed throughout the file, you may wish to increase the amount of CI free space and decrease the amount of CA free space.
  - To do this, FRSPCI and FRSPCA may be used. However, make sure you compare results, before and after, to see what other recommendations may have changed. The best way to determine that this override is required after file tuning, is the appearance of excessive CI splits during normal monitoring.
- When the insert activity to the file is such that inserts occur in pockets around the file, you may wish to decrease the amount of CI free space and increase the amount of CA free space.
  - To do this, FRSPCI and FRSPCA may be used. However, make sure you compare results, before and after, to see what other recommendations may have changed. This kind of insert activity tends to be more common. The best way to determine that this override is required after file tuning, is the appearance of excessive CA splits during normal monitoring.

If you decide to override CBLVCAT's free space recommendations, without the use of the influencing sub-parameters, make sure you fully understand the consequences. Incorrect choices will lead to any combination of dead and wasted space, poor performance, excessive CI splits and especially excessive CA splits.

If you decide to adopt CBLVCAT's recommended free space values, you do not have to wait until the file is backed up, re-defined and re-loaded. You can use the IDCAMS ALTER command to update the catalog entry immediately. This does not mean, however, that the free space attributes will become immediately effective. The file still needs to be reloaded for the free space percentages to take effect. Nevertheless, during split processing VSAM does apply freespace values generated by the IDCAMS ALTER command.

**SPLITS**

A Control Interval split occurs if there is insufficient free space available in the CI for a new record (or to extend an existing record). One of the free CI's within the same CA will be used.

A Control Area split occurs if there are no free CI's within a CA. The new CA is allocated at the end of the existing allocated space, otherwise secondary allocation takes place and the split CA is placed in the secondary extent.

The rules for CI SPLITS and CA SPLITS are as follows:-

**Sequential Processing**

**CI SPLIT**

If the insert is in the middle of the CI, the records with higher keys are moved to the free CI. The insert and the records with lower keys remain in the old CI. If the insert is at the logical end of the CI, the inserted record goes to the free CI.
CA SPLIT
If the insert is not in the last logical CI, all CIs after the split CI are moved to the new CA. If the insert is within the last logical CI, that CI is moved to the new CA. If the insert is at the end of the last logical CI, the inserted record is placed into the new CA.

Direct Processing
CI SPLIT
Half the records (those with the higher keys) in the CI are moved into the new CI. The new record is inserted (in key sequence) into the CI to which it belongs.

CA SPLIT
Half the CIs (those with the higher keys) are moved to the new CA. Insertion then occurs through regular CI split processing, using the newly created free space CIs.

Single/Mass Insertion
This is a technique automatically used by VSAM when, for instance, a file is opened for output. Mass insertion reserves defined FREESPACE and does not perform CI/CA splits. Input records must therefore be sorted in ascending key value.

Performance
CA splits can cause severe system or terminal response degradation
Reloading the file removes CI and CA splits with an inevitable improvement in DASD usage and response time.

KEY COMPRESSION
Within the index component, and whenever possible, VSAM will automatically compress a key in order to minimise space used.

Characters from both the front and back of a key are removed if they are redundant when comparing the current entry key with the key immediately before and after it.

The following key sequences would be compressed:

```
12345ABCDE67890
12345ABCDE67891
12345ABCDE67892
12345ABCDE67893
12345ABCDE67894
```

The following key sequence would not be compressed:

```
12345ABCDE67890
22345ABCDE67891
32345ABCDE67892
42345ABCDE67893
52345ABCDE67894
```

ALTERNATE INDEX and PATH

AIX
An alternate index (or alternate indices) can be built by VSAM over a KSDS or ESDS dataset.

An alternate index allows the base cluster to be accessed via an alternate key.

Alternate keys, unlike the prime key of a KSDS, do not have to contain unique values.

The base cluster has to be defined and loaded for the alternate index to be defined and built via AMS commands.

VSAM can be requested to maintain the alternate index so that any changes to the base cluster are reflected in the alternate index. The AIX is then part of the UPGRADE SET. CBLVCAT shows this by displaying as AIX (U).

The data portion of the AIX consists of:-

- 5 bytes of system header information.
- The alternate key
- At least one prime key

<table>
<thead>
<tr>
<th>SYSTEM INFORMATION</th>
<th>ALTERNATE KEY</th>
<th>PRIME KEYS</th>
<th>.................</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2013-10-29 15:43:01 CBLVCAT Manual 170
If the AIX is opened the data portion above is read.

**PATH**

To access a base cluster through the alternate index a PATH must be defined.

Referencing the PATH name causes the ALTERNATE INDEX and its related base cluster to be referenced. Opening the PATH entry name causes the base cluster to be processed using the alternate key.

The PATH relates to, or sets up, an association between the AIX and the BASE and is where the **UPDATE/NOUPDATE** option is made.

**Performance**

An AIX should be treated as a KSDS.

Alternate indices, which are part of the upgrade set, must be updated to reflect changes to the base cluster. This will obviously cause additions/deletions to take longer to process than would otherwise be the case.

If a path is opened, the upgrade set will be maintained if the path has been defined as **UPDATE**. If the path has the attribute **NOUPDATE** the upgrade set will not be maintained.

Extra bufferspace (i.e. more than 2 data buffers and 1 index buffer) can be allocated to an alternate index and will be used if the access is through the alternate index. If the alternate index is being processed as part of the upgrade set, the minimum buffer space will be used.

**LEVELS OF INDEX**

In each record of a KSDS, there is a fixed length, fixed position field that contains a unique value for each record. This is the key field on which the index is built.

The index consists of :-

- The sequence set (the lowest level).
- The index set (all other levels).

The sequence set index record contains an entry for each CI in its related CA. The entries hold the highest key (compressed) that can be held on that CI, plus a pointer to that CI.

The higher levels of index in the index set, are built as required by VSAM.
When accessing the index, only one CI at each level is accessed. VSAM will therefore build further levels of the index until the highest level consists of a single CI.

Factors that govern the number of index levels include:

- Key length
- Key compression
- Freespace
- Index cisize

When Index CISIZE is left to VSAM, the calculated index CISIZE will be large enough to hold as many entries as there are data CIs in a CA (Beware of large keys as the anticipated compression may not materialise).

At least two levels of index will be required if the file occupies more than one data CA.

BUFFER SPACE

Buffer Space (BUFSP) is a quantity of computer memory used by VSAM for the transfer of records (in units of Control intervals), between disk and computer storage. The amount of reserved space can be specified in three ways:

1. In the catalog when the file is defined.
   This will be the minimum amount of storage which can be reserved. The program ACB, or JCL, can be used to increase the amount of buffer space, but never to decrease it.

2. In the program, via the ACB (Access Method Control Block).
   This describes the file’s characteristics to VSAM. The buffer requirements must result in a BUFSP value which is greater than or equal to the value specified in the catalog. If this is not the case, then the catalog value will be used instead.

3. In the operating system JCL.
   This is the Job Control Language of the operating system. The buffer space requirements specified override the value specified in the program’s ACB, but must be greater than or equal to the value specified in the catalog. If this is not the case, then the catalog value will be used instead.

The optimum amount of buffer space VSAM is to use depends mostly on how the file will be processed (i.e. sequentially or randomly), and the number of index levels required to point to the data.

For all files, CBLVCAT will recommend a BUFSP value which is best for direct processing.

BUFSP for Sequential Processing

For ESDS and RRDS files, all buffer space by definition is used for data.

For a KSDS, space for just one index buffer is required by VSAM and the remaining buffer space is used for data buffers.

Any additional data buffers available are used by VSAM in two ways:

1. By reading more than one control interval at a time.
   i.e. by building a CCW chain to read several CIs with one I/O initiation (or EXCP).

2. By initiating the next read in advance.
   i.e. when data is returned to the program by VSAM after an EXCP, the next EXCP is issued immediately. This is so that the next EXCP can take place whilst the program is dealing with the data buffers filled by the first EXCP.
   This uses a VSAM feature called Lookahead Read.

It should be noted that different versions of VSAM give different priorities to these 2 buffering techniques.

Sequential processing is significantly improved by using a JCL override on the BUFSP value sufficient for several data buffers, plus 1 index buffer.

The override of BUFSP for sequential processing is achieved via JCL.

Pre VSE/SP 4.1 users should code BUFSP=nnnn on the DLBL statement. If there are less than 4 data buffers available, VSAM will use I/O command chaining to fill all the buffers and then return to the program. When 4 or more data buffers are available, VSAM will fill two or more buffers, return control to the program and initiate the I/O operation to fill the remaining buffers. Thus, Lookahead Read is achieved. Use of more than four or five data buffers should be evaluated carefully in relation to the demanding effect it will have on the rest of the system.

MVS users and VSE users from VSE/SP 4.1 have the advantage of being able to supply BUFND=5 on the DD statement (DLBL for VSE) instead. This avoids the need to calculate or transcribe large numbers.

The advantage of defining a large BUFSP for sequential programming, however, has to be balanced against possible extra virtual storage demands, paging, and CPU monopolisation.

BUFSP for Direct Processing
If you decide to implement CBLVCAT's recommended BUFSP value to define the file, the BUFSP value is correct for random processing of KSDS files.

No further special action is required for programs which access KSDS files randomly.

For ESDS and RRDS files, no index exists, so the buffer space reserved is only for 2 Data CIs, which will suffice for good performance.

Additional buffer space should be specified via JCL overrides and will be used to accommodate additional Data CIs.

ESDS files, which are sequential in nature, can be processed by programs in random fashion through the use of RBA (Relative Byte Address).

RRDS files, which are random in nature, can be processed sequentially. In light of this, you should use JCL to specify additional buffers as required. Otherwise, you can adjust the DEFINE value to suit the mode of processing when it is always the same.

For KSDS files, VSAM requires two data buffers, and a minimum of one index buffer. For efficient processing, 1 index buffer per level of index plus 1 is required.

If additional buffer space is available, VSAM uses it to accommodate more index CIs. While this may save the occasional I/O for an index CI already in storage, it is possible to cause an undesired increase in CPU usage due to increased storage demands. It also results in longer instruction path lengths which are required to look for the index record, regardless of whether it is in storage or not.

Bear in mind that updates will require a refresh of buffers.

The KSDS index has a hierarchical organisation expressed in terms of index levels. The top level always contains one CI. The bottom level is called the Sequence Set and consists of one record per data control area, normally a cylinder or MAX-CA.

Thus, all but small KSDS files will have at least two index levels (i.e. files occupying more than one control area). Each level consists of pointers to the next level down. As soon as the top level index requires more pointers than it is able to hold in one CI, it is split into 2 CIs and a new top level of index is created, initially having only 2 pointers. And so it goes on.

It is extremely important to ensure that there is at least one index buffer per level of index when a KSDS is processed randomly.

Consider a large file which has 275 index records spread over three levels and where the data resides in 220 control areas. The bottom index level (the sequence set) contains 220 records (one per data control area), the top level contains 1 (by definition) and so the second, or middle level contains the remaining 54.

If there is only one index buffer, three index reads will always be required per record retrieved (one per level). This must be avoided.

If there are three index buffers, the top level index is read once only and the number of index reads per retrieval ranges from 0 to 2, depending on the contents of the buffers for the other two index levels. Usually it will be 2.

Increasing buffer space for the above example will result in more of the second level available in storage, giving the advantage of reducing index reads. However, the penalty of the additional CPU overhead, due to checking if an index record is in storage, is incurred whether the index record is found or not. This can outweigh all the advantages.

Use of 56 index buffers would allow 1 for the top level, 54 for the 2nd level, and 1 for the bottom level. This would result in 0 or 1 index read per record retrieved. Usually 1. But is the trade-off in CPU time and storage cost worth it?

CBLVCAT's compromise is to cater for 1 extra index level, which allows a jump to the next index level after further inserts, without degradation of efficiency.

**BUFSP for Sequential and Direct Processing**

When files are opened for sequential and direct processing, VSAM will default to using the buffer space in the same way as it does for sequential processing (i.e. 1 index buffer and the remainder for data buffers).

If processing is predominantly sequential then this is satisfactory. Benefit may be gained by increasing the buffer space to allow another data buffer, however, the look-ahead sequential reads are wasted every time a direct read occurs. As we already have 2 data buffers, there is doubt as to whether this change would be of value.

If processing is predominantly Direct then VSAM’s assumption is wrong. VSAM will default to one index buffer only and so, without an override, you will not get maximum efficiency.

For MVS, the required override is to increase the number of index buffers to (No. of index levels)+1, using BUFNI=n in the JCL.

Buffer space should not be modified or adjusted because the definition, which should be in the catalog for this file, is either correct, or will be recommended correctly for predominantly direct processing.

BUFND should not be modified because VSAM will use the remainder as data buffers anyway.

Obeying the BUFNI = (No. of index levels)+1, VSAM will then operate at maximum efficiency for direct processing.

For pre VSE/SP 4.1, the override of BUFNI is not currently supported within its JCL, so the inefficiency should be corrected by introducing the required BUFNI = (No. of index levels)+1 into the program's ACB.
The SPEED and RECOVERY options allow the user to specify how much preformatting is done when a file is loaded.

RECOVERY

RECOVERY will cause VSAM to preformat a control area in advance and to write an EOF record on the CA. If the load fails, AMS Verify can be used to locate the EOF record and the rest of the records can be added in extend mode. This assumes that the loading program is smart enough to continue where it left off.

IDCAMS REPRO and most other load programs do not offer a restart facility.

SPEED

No preformatting is done. If the load fails, all the records will have to be reloaded, after the file has been deleted and re-defined.

Performance

SPEED only has an effect on performance at load time. After the file has been loaded, all future extensions will be done in recovery mode.
Acronym | Definition
--- | ---
ACB | Access Method Control Block. (VSAM)
AIX | Alternate Index (see Additional VSAM Information).
ALIAS | An alternate name for a Catalog or NONVSAM file (OS systems).
AMS | Access Method Services.
ASSOC | File ASSOCiation (AIX, PATH etc).
BCS | Basic Catalog Structure (part of ICF Catalog).
BLKSIZE | The physical size of records.
CA | Control Area. A number of Control Intervals are grouped together to make up a Control Area. (see Additional VSAM Information).
CAP Block | CBLVCAT tuning capacity block.
CATALOG | The VSAM directory.
CI | Control Interval. A VSAM storage unit to contain records, similar to a block in other access methods. It also contains control information which VSAM uses and which you do not see. (see Additional VSAM Information).
CI/CA | The number of Control Intervals in a Control Area.
CICS | Customer Information Control System. IBM software program.
CISIZE | Control Interval SIZE.
CKD | Count Key Data. A type of disk architecture.
CLUSTER | VSAM maintains files in structures called CLUSTERS. A CLUSTER is maintained on disk in one of two ways: - ◊ For non-ICF environments the CLUSTER lives in an area of disk reserved for VSAM called a VSAM DATA SPACE. The clusters within the SPACE are under control of a USER CATALOG and a MASTER CATALOG.
◊ For ICF environments the CLUSTERS are under the control of an ICF Catalog which is composed of two parts, the BCS and the VVDS. The VVDS contains the information such as file characteristics. There is no concept of VSAM SPACE in this environment. See the relevant IBM documentation for more information.
CMS | Conversational Monitoring System. Operating System component of VM.
CMS/VSAM | IBM software program.
CRA | Catalog Recovery Area.
DA | Direct Access.
DASD | Direct Access Storage Device. A disk.
DBD  Data Base Descriptor

DF/EF  Data Facility Extended Function
       Used with ICF Catalogs.

DDNAME  'Data Set label' in MVS systems.

DLBL  'Disk label information' in VSE systems.

DL/I  Data Language/1
       Hierarchical database system for VSE (c/f IMS for MVS).

DOS  Disk Operating System

DOS/VS  Disk Operating System/Virtual Storage

DOS/VSE  Disk Operating System/Virtual Storage Extended.

EOF  End of File

ESA  Enterprise System Architecture

ESDS  Entry Sequence DataSet
       A VSAM file structure analogous to sequential files. ESDS files can be processed sequentially from start to end, or records
       can be accessed directly by supplying the Relative Byte Address (RBA) of the record.

EXEC  A CMS filetype used to store a set of statements.

EXCPS  EXecute Channel Programs (I/O)

FBA  Fixed Block Architecture
       A type of disk architecture.

FCT  File Control Table
       A component of CICS.

FILEDEF  CMS terminology for 'file label'.

GDG  Generation Data Group

ICF  Integrated Catalog Facility
       A Catalog consists of a BCS and one or more VVDS.

ICFCAT  VSAM Catalog on ICF

IDCAMS  IBM utility program for VSAM Catalogs.

IJSYSCT  VSE DLBL for the VSAM Master Catalog.

IJSYSUC  VSE DLBL for the VSAM Job Catalog.

IMS  Information Management System
       Used to mean IMS/DB. A hierarchical database for MVS (c/f DL/I for VSE).

INDEX  The index extent of a VSAM/ISAM file.
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISAM</td>
<td>Index Sequential Access Method</td>
</tr>
<tr>
<td>IX</td>
<td>IndeX component of AIX or KSDS file.</td>
</tr>
<tr>
<td>IXL</td>
<td>IndeX Levels (see Additional VSAM Information).</td>
</tr>
<tr>
<td>JCL</td>
<td>Job Control Language</td>
</tr>
<tr>
<td>JCL Block</td>
<td>CBLVCAT tuning JCL block.</td>
</tr>
<tr>
<td>JOBCAT</td>
<td>MVS Job Catalog.</td>
</tr>
<tr>
<td>KSDS</td>
<td>Key Sequence DataSet. A VSAM file structure made up of 2 components, an INDEX component and a DATA component. Each component is a separate file.</td>
</tr>
<tr>
<td>LDS</td>
<td>Linear DataSets. Consists of a long stream of bytes and not considered to have records.</td>
</tr>
<tr>
<td>MASTCAT</td>
<td>VSAM master Catalog</td>
</tr>
<tr>
<td>MAX-CA</td>
<td>MAXimum Control Area size (see Additional VSAM Information).</td>
</tr>
<tr>
<td>MIN-CA</td>
<td>MINimum Control Area size (see Additional VSAM Information).</td>
</tr>
<tr>
<td>MVS</td>
<td>Multiple Virtual Storage (operating system).</td>
</tr>
<tr>
<td>MVS systems</td>
<td>Generic term for OS, OS/VS, MVS, MVS/XA, MVS/ESA and OS/390.</td>
</tr>
<tr>
<td>NONVSAM</td>
<td>MVS files which do not occupy space within the Catalog.</td>
</tr>
<tr>
<td>NSPND</td>
<td>Non SPaNneD file.</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System.</td>
</tr>
<tr>
<td>OS/VS</td>
<td>Operating System/Virtual Storage.</td>
</tr>
<tr>
<td>OS/390</td>
<td>Open integrated Server Operating Sytem Environment. The latest release of MVS.</td>
</tr>
<tr>
<td>PAGESP</td>
<td>MVS system PAGE SPace.</td>
</tr>
<tr>
<td>PRIME</td>
<td>The PRIME data extent of VSAM or ISAM file.</td>
</tr>
<tr>
<td>RBA</td>
<td>Relative Byte Address.</td>
</tr>
<tr>
<td>RDF</td>
<td>Record Descriptor Field.</td>
</tr>
<tr>
<td>REORG</td>
<td>A file reorganisation.</td>
</tr>
<tr>
<td>REPRO</td>
<td>An IBM utility program.</td>
</tr>
<tr>
<td>RRDS</td>
<td></td>
</tr>
</tbody>
</table>
Relative Record DataSet.
A VSAM file structure having a "pigeon hole" or slot structure. Each slot is associated with a Relative Record Number. It can be processed sequentially, or randomly by its relative record number.

SAM
Sequential Access Method

SEOF
Software End Of File.

SEV Block
CBLVCAT tuning block containing SEVerity messages.

SMS
System Managed Storage

STACK
A CMS area used to pass data between commands and programs.

STANDARD LABEL AREA
An area of DASD used to store label information.

STECAT
MVS description of Catalog to be used for a job step.

SPANDED
Records which span control intervals.

SPLIT
Action taken on a CI or CA for some record insertions (VSAM), or a file occupying a split cylinder (VTOC).

STRNO
Number of strings used.

SYSIN
SYStem INput unit for OS/MVS and CMS.

SYSIPT
SYStem InPuT unit for VSE.

SYSLST
SYStem LiST unit for VSE.

SYSPCH
SYStem PunCH unit for VSE.

SYSPRINT
SYStem PRINT unit for MVS.

SYSPUNCH
SYStem PUNCH unit for MVS.

TSO
Time Sharing Option

TUNE Block
CBLVCAT tuning block containing IDCAMS define recommendations.

USERCAT
VSAM USER CATalog.

VERIFY
IDCAMS command to reset a file's high used RBA.

VSAM
Virtual Storage Access Method

VSE
Virtual Storage Extended

VSE systems
Generic term for DOS, DOS/VS, VSE and VSE/ESA.

VSE/ESA
Virtual Storage Extended/Enterprise System Architecture

VTOC
Volume Table Of Contents

VVDS
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>VSAM Volume Data Set</td>
<td>(used with BCS)</td>
</tr>
<tr>
<td>VVR</td>
<td>VSAM Volume Record (For a VVDS)</td>
</tr>
<tr>
<td>XA</td>
<td>EXtended Architecture (MVS)</td>
</tr>
</tbody>
</table>
SELCOPY the Productivity Aid

From its inception as a SELect and COPY utility, SELCOPY has developed into an all purpose productivity aid, for MVS, VM and VSE users.

See the CBL web site for more detailed information on SELCOPY.

The Multiple Utility
The same simple free format control cards rationalize all file to file utilities under one program. Capable not only of multiple input, output and printing, but also selective modification, and all at the same time. Conversion includes ASCII, EBCDIC and Hex representation as well as arithmetic.

Printing choice includes Char, Hex, Both, Mixed, Dump, or Report. Alternatively, you may convert some fields, rearrange them, and print a subset.

The Trouble Shooter
Simple, quickly written code will scan files for problem records. Having checked the scope of the errors, equally simple code can be used to correct the files.

File Format Conversion
Changing Fixed, Variable, Undefined and reblocking is just the beginning. SELCOPY may be used to restructure any non-IBM file into standard IBM format.

Test Data Generator
Generated data is controlled at the field level, with range controls on all data types in conjunction with all SELCOPY’s selection logic.

Test files may thus be generated from nothing, or from a controlled selection of records from existing files with controlled regeneration of selected fields.

VSAM Usage
SELCOPY contains the full complement of VSAM facilities, such as Update in place, Keyed Read, Sequential Read, (both backwards & forwards), Insert and Delete. Such facilities are coupled with SELCOPY logic and its variety of input and output files, using the same simple syntax.

Back-Up and Restore
During your regular back-up run, you can code SELCOPY to give a useful small report on what it has backed up. Print the number of each type of record for example.

SELCOPY can back-up many totally separate files, VSAM or other, even off different volumes, concatenating them together to one single file on tape, while still keeping it logically processable.

The VM/CMS Environment
Read/Write/Update CMS files with native CMS I/O (FSREAD and FSWRITE). Read Sequential files and Read/Write/Update VSAM files, on VSE and MVS disks linked to your CMS id.

Unique to SELCOPY is its Keyed Read for CMS on files which are in sequence on some field within the record (RECFM V as well as F).

Multi-File Scanning
Scan all members of an MVS PDS, searching for a string, report and optionally modify and update-in-place. Input could equally be a generic group of VSE library members or CMS files, but for VSE, a second step is required to submit updates to LIBR.

ADABAS
SELCOPY supports the ADABAS database.

IMS and DL1 Usage
Insert, Delete, Replace, Sequential Read, Qualified Read, with full DL1 syntax for qualifiers, is provided with the customary SELCOPY simplicity.

The TSO Environment
Just ALLOC for your files and use the full range of SELCOPY’s MVS facilities. Run conversationally using WTO with the REPLY function. For standard procedures, run from CLISTs or REXX allowing variables to be passed to SELCOPY on the invoking command line.

Speed of Execution
SELCOPY is written in Assembler, so is able to minimise linkage to I/O routines, but still uses standard IBM Data Management. This has the double advantage that it maximises speed while keeping up-to-date with IBM development, and at the same time avoids channel monopoly locking out other users.