

Unify ACCELL/SQL

機能拡張ガイド

Releases 7.1 - 7.2

© 2000, 2001 Unify Corporation. All rights reserved.

Publications team Natalie Calkins
Linda Costello

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise without the prior written consent of Unify Corporation.

Unify Corporation makes no representations or warranties with respect to the contents of this document and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Further, Unify Corporation reserves the right to revise this document and to make changes from time to time in its content without being obligated to notify any person of such revisions or changes.

The Software described in this document is furnished under a Software License Agreement. The Software may be used or copied only in accordance with the terms of the license agreement. It is against the law to copy the Software on tape, disk, or any other medium for any purpose other than that described in the license agreement.

Unify, ACCELL, and Unify VISION are registered trademarks or Unify Corporation in the United States and other countries. Unify DataServer, Unify eWave, Unify WebNow!, and the Unify logo are trademarks of Unify Corporation in the United States and other countries.

Sun, Sun Microsystems, the Sun Logo, Java, JavaBeans, Enterprise JavaBeans, JavaServer Pages, Java Naming and Directory Interface, J2EE, JDBC, and JDK are trademarks or registered trademarks of Sun Microsystems, Inc.

Macromedia, the Macromedia logo, Dreamweaver, and UltraDev are trademarks or registered trademarks of Macromedia, Inc.

Linux is a registered trademark of Linus Torvalds.

Red Hat, the Red Hat "Shadow Man" logo, RPM, and the RPM logo are trademarks or registered trademarks of Red Hat, Inc.

Caldera Systems, the C-logo, and OpenLinux are trademarks or registered trademarks of Caldera Systems, Inc.

SuSE and its logo are registered trademarks of SuSE AG.

Microsoft, Windows 95, Windows 98, Windows NT, and Windows 2000 are trademarks or registered trademarks of Microsoft Corporation.

Intel and Pentium are trademarks or registered trademarks of Intel Corporation.

Other brand or product names shown are trademarks of their respective owners.

Part Number:7205-01

目次

このマニュアルについて.....	4
対象とする読者	4
表記ルール.....	4
テクニカルサポート	5
第 1 章 データ入力時の文字チェック.....	7
USTRBCHARS コンフィギュレーション変数	7
UTXTBCHARS コンフィギュレーション変数.....	9
第 2 章 AMGR 終了時の Status	10
第 3 章 日付フォーマット.....	11
セパレータ無しでの DATE 値の入力	11
曖昧な日付に対する新しいコンパイラ警告	12
WITH NO FORMAT を指定するときに 4 桁年を上書きする....	12
2 桁の年の日付	12
第 4 章 RPT - 数値入力フィールドでの警告メッセージ	13
第 5 章 String または Text からの変換	16

このマニュアルについて

このマニュアル (ACCELL/SQL: *機能拡張ガイド Release 7.1- 7.2*) は、ACCELL/SQL リリース 7.1、7.2 において追加された新機能について説明します。このマニュアル内の情報は既存の ACCELL/SQL のマニュアル (Product Information Supplement: Third printing of the ACCELL/SQL manual set (7295-03) を含む) の補足となります。

対象とする読者

このマニュアルは ACCELL/SQL を使用する開発者のために書かれています。

表記ルール

このマニュアルでは、次のような表記方法を使っています。

通常の記事

< >

不等記号で囲まれた語は、引数、変数、数値、または提供すべき式などを表します。

Monospaced フォント

必ず入力しなければならないコードサンプルやテキストはこのフォントで表されています。

Windows

特に指定のない限り、Windows は Windows NT と Window2000 の両方を意味します。

特に重要な段落

ヒント

このマークのついた段落には、推奨する操作などの有用な情報が含まれています。

要注意

このマークのついた段落には、データの損失を引き起こす可能性のある操作について警告しています。

関連情報

このマークのついた段落には、そのセクションとトピックに関連した参照情報を提供します。

テクニカルサポート

ユニファイ製品に関して何か問題が起こった場合は、色々な情報が提供されている Web サイト上のテクニカルサポートをご覧ください。ユニファイのテクニカルサポートには下記のアドレスよりアクセスできます。

<http://support.unify.com/supportal/Home.html>

テクニカルサポートが必要な場合には、まず、ユニファイテクニカルサポート FAQ に探している答え (情報) があるかチェックしてください。ユニファイテクニカルサポート FAQ には下記のアドレスよりアクセスできます。

<http://support.unify.com/supportal/FAQs.htm>

FAQ を見ても問題が解決しない時には、ユニファイカスタマの皆さんによるディスカッションフォーラムであるメーリングリストへの参加をおすすめします。メーリングリストを購読するには、下記の電子メールアドレス宛てに、

listserv@unify.com

次のようなテキストをメッセージ本文に入力して、メールを送信してください。

```
subscribe customers
```

メールの受信後、確認のメールをお送りします。このメールには、メーリングリストサービスへのメールでの質問の仕方や登録の解除方法も書かれています。

ユニファイ製品ドキュメントの Web サイトではリリースされてからのドキュメントの変更点を確認できます。また最新のバージョンのドキュメントをダウンロードすることもできます。Unify 製品ドキュメントの Web サイトには次の URL よりアクセスできます。

<http://support.unify.com/supportal/Docs.htm>

第 1 章

データ入力時の文字チェック

リリース 7.2 より利用可能

2つの新しいコンフィギュレーション変数、USTRBCHARS と UTXTBCHARS を設定することにより、ユーザが STRING および TEXT フィールドに不必要な文字を誤って入力することを避けることができます。

ユーザはキーボードのキー配列が原因で、テキストフィールドに思いがけなく文字を入力してしまうことがあります。不必要な文字、たとえばバックスラッシュなど、入力してもデータ値の一部として処理されます。不要な文字があると想定しないので、そのフィールドの検索は、それ以降失敗することになります。加えて、データを調査し、不要な文字を探し出すことは難しく、問題を判別することを困難にします。

コンフィギュレーション変数を使って、アプリケーションの STRING、TEXT のデータ型、あるいは両方の全てのテキストフィールドで入力時に拒絶される文字のリストを指定することができます。

USTRBCHARS コンフィギュレーション変数

USTRBCHARS コンフィギュレーション変数は、データ型 STRING のテキストフィールドにデータ入力（追加 / 更新モード）からブロックされる文字をリスト化します。

ユーザが不必要な文字を入力しようとするとき、ビーブ音が聞こえ、文字は受け入れられません。GUI モードでは、NEXT FIELD のようなデータ処理コマンドを実行するまでは入力された文字がチェックされない点に注意してください。

文字列がフィールドに貼り付けられるとき、1つ以上の不要な文字が文字列に存在する可能性があります。GUI モードでは、ペーストされた文字のセット全てがブロックされ、エラーメッセージは最初の不要な文字を見つけ識別します。キャラクターモードでは、不要な文字だけがブロックされ、有効な文字はフィールドで許可されます。

以下のテーブルは USTRBCHARS コンフィギュレーション変数の要約です。

依存度 : なし
有効な値 : キャラクター表現の文字列はタブやスペースによって分けられていません。

印刷できる ASCII 文字（下のリストの例外を除く）はそれ自身かあるいは 8 進、16 進値で表現されます。NULL(0) 以外の印刷できない文字はすべて、8 進、16 進値で表現されなければなりません。NULL は許可されていません。

8 進値は、0 で始まる 01 ~ 0377 の範囲の値でなければなりません。
16 進値は、0x で始まる 0x01 ~ 0xff の範囲の値でなければなりません。

16 進定数のアルファベットの桁 (a ~ f) は、大文字、小文字のどちらでもかまいません。

以下の 4 つの印刷可能な文字は、8 進、16 進値を使用して表現されなければなりません。

- アポストロフィ (一重引用符)
047 (8 進値) 0x27 (16 進値)
- ダブルクォーテーション (二重引用符)
042 (8 進値) 0x22 (16 進値)
- バックスラッシュ (/)
0134 (8 進値) 0x5C (16 進値)
- 曲折アクセント (^ハット)
0136 (8 進値) 0x5E (16 進値)

文字列に不要な文字があった場合、すべてのセッティングが無視されます。

デフォルト値 : なし (アプリケーション内で STRING 型のテキストフィールドとして認められない文字はありません)

例 : @、*、/、' これらの文字をブロックするには、次のように設定します。

```
USTRBCHARS = "@ * 0134 0x27"
```

追加情報 : TEXT 型のテキストフィールドで文字をブロックするには、[「UTXTBCHARS コンフィギュレーション変数」](#)を参照してください。

UTXTBCHARS コンフィギュレーション変数

UTXTBCHARS コンフィギュレーション変数は、データ型が TEXT のテキストフィールドに対するデータ入力 (追加 / 更新モード) からブロックされる文字をリストします。

ユーザが不要な文字を入力しようとする、ビープ音がなり文字は受け付けられません。GUI モードでは NEXT FIELD のようなデータ処理コマンドを実行するまでは、入力された文字がチェックされない点に注意してください。

文字列がフィールドに貼り付けられるとき、1 つ以上の不要な文字が文字列に存在する可能性があります。GUI モードでは、ペーストされた文字のセット全てがブロックされ、エラーメッセージは最初の不要な文字を見つけ識別します。キャラクターモードでは、不要な文字だけがブロックされ、有効な文字はフィールドで許可されます。

依存度 :	なし
有効な値 :	キャラクター表現の文字列はタブやスペースによって分けられています。詳細は、USTRBCHARS コンフィギュレーション変数を参照してください。
デフォルト値 :	なし (アプリケーション内で STRING 型のテキストフィールドのために認められない文字はありません)
例 :	@、*、/、' これらの文字をブロックするには、次のように設定します。 UTXTBCHARS = "@ * 0134 0x27"
追加情報 :	STRING 型のテキストフィールドで文字をブロックするには、 「USTRBCHARS コンフィギュレーション変数」 を参照してください。

第 2 章

AMGR 終了時の Status

リリース 7.2 より利用可能

セッションが致命的エラーのためにアボートする場合、プログラムが AMGR の終了時 Status を判別できるようになりました。

AMGR の終了 status がゼロ以外の場合、致命的エラーが発生します。但し、終了 Status はエラーの詳細な情報は識別しません。詳細な情報は AMGR による出力、もしくは表示されるエラーメッセージを参照することにより識別されます。致命的エラーは、AMGR Error Messages セクションで記述されます。

第 3 章

日付フォーマット

リリース 7.1 より利用可能

セパレータ無しでの DATE 値の入力

以下の場合には、日付データを区切り文字なしで入力することができます。

- 1 月、日に対して 2 桁の数字を入力する。
- 2 DATEFMT で指定されている順序で、日付データの月、日、年を入力する。
- 3 2 桁の年を入力する場合には、CENTURY_CUTOFF を適切な値に設定する必要があります。デフォルト値は 00 ですので、2 桁の年は、1900 年から 1999 年の間とみなされます。
- 4 DATEFMT がアルファベットが月として設定されていても、2 桁の数字の月を入力する。

DATEFMT で 2 桁の年 (YY/MM/DD) が指定されていても、4 桁の年を入力できます。また、DATEFMT で 4 桁の年 (DD.MM.YYYY) が指定されていても、2 桁の年を入力できます。

DATEFMT で定義される区切り文字は、検索範囲を表すメタ文字と異ならなければなりません。

DATEFMT の設定と有効な入力の例を下記に示します。

DATEFMT 値	有効な日付入力
YYYY MM DD	991210
YY/DD/MM	19991012
DD.MM.YYYY	101299

曖昧な日付に対する新しいコンパイラ警告

ACPL、disc などの Unify コンパイラ、およびストアードプロシージャやトリガ用のコンパイラは、処理する日付の年号が曖昧な場合、警告メッセージを表示します。例えば、2/2/50 の日付は、2/2/1950 と 2/2/2050 のどちらにも取れます。

WITH NO FORMAT を指定するときに 4 桁年を上書きする

SQL での WITH NO FORMAT オプションは、可能な最大精度を付けてデータを表示します。デフォルトでは 4 桁の年を強制します。この動作を上書きして 2 桁表示にするには、コンフィギュレーション変数 WF2DIGITYEARS を TRUE に設定します。デフォルトの設定は FALSE になっています。

WF2DIGITYEARS を TRUE に設定する場合、数字の桁数は WITH FORMAT が指定されるように DATEFMT で定義されます。

2 桁の年の日付

2 桁の年の日付が、CENTURY_CUTOFF で指定された範囲外である場合、混乱を招くことがあります。この混乱を避けるため、Accell マネジャはこの範囲外の日付をすべて 4 桁の年を使って表示するようになりました。

例えば、CENTURY_CUTOFF が 25 の場合、1925-2024 の 100 年間を定義します。つまり、この範囲外の日付は、2 桁の年を使って画面に表示されます (1/1/25 = 1/1/1925; 3/1/24 = 3/1/2024)。しかし、この 100 年間の範囲外の日付は、DATEFMT の設定にかかわらず、常に 4 桁の年を使って表示されます (DATEFMT=MM/DD/YY の場合でも 1/1/1924 = 01/01/1924)。

第 4 章

RPT - 数値入力フィールドでの警告メッセージ

リリース 7.1 より利用可能です。

数値入力フィールドに 9 よりも大きな長さが指定された場合、以下の警告メッセージが stderr に書き込まれます。(スクリプト・ファイルの 2 行目に、MyField [numeric 11] が指定された場合)

```
input
  foo [numeric 11],
  berf [numeric 10]
default
  print 'hi'

end
```

"RPT スクリプト名 " を実行した場合、以下のメッセージが stderr に書き込まれます。

```
RPT - Report Processor
Copyright Unify Corporation 1983, 1984, 1985, 1988.
```

```
-----
Warning: [ numeric 11 ] exceeds the maximum length of this type (9)
on or about line 2
Use a float type if you expect more than 9 digits in this field.
```

```
-----
Warning: [ numeric 10 ] exceeds the maximum length of this type (9)
on or about line 3
Use a float type if you expect more than 9 digits in this field.
```

```
-----
syntax error
on or about line 4
```

```
-----
-----
There is an invalid input item (-10209)
on or about line 4
```

```
-----
Syntax error(s) were found in the report script.
```

RPT テーブル使用状況

テーブル名	Used	最大値
式ノード	0	400
変数	2	150
定数	0	250
コマンド	0	256
プリント 命令文	0	125
プリント 項目	0	256
ソート 項目	0	15
入力 項目	2	100
コマンドグループ	0	25
セット 命令文	0	100
If 命令文	0	50
総計	0	50
関数コール	0	50
引数	0	100

入力データがデータファイルで提供されないとき、RPT テーブル使用状況が出力されます。入力データがデータファイルで提供されるとき、RPT テーブル使用状況は出力されず、以下のメッセージが提供されます。

```
$ RPT rptscpt mydata
```

```
-----  
Warning: [ numeric 11 ] exceeds the maximum length of this type (9)  
on or about line 2  
Use a float type if you expect more than 9 digits in this field.
```

```
-----  
-----  
Warning: [ numeric 10 ] exceeds the maximum length of this type (9)  
on or about line 3  
Use a float type if you expect more than 9 digits in this field.
```

```
-----  
-----  
syntax error  
on or about line 4
```

```
-----  
-----  
There is an invalid input item (-10209)  
on or about line 4
```

```
-----  
Syntax error(s) were found in the report script.
```

第 5 章

String または Text からの変換

リリース 7.1 より利用可能

`str_to_val$()` 関数に対して可能な入力引数は、STRING 型と TEXT 型です。

関連情報

`str_to_val$()` システム機能の完全な詳細は、ACCELL/SQL マニュアル "System Functions and Variables" を参照してください。

入力引数に使用できる文字は以下に限られます。

- 先頭空白文字
- プラス記号 (+) とマイナス記号 (-)
- 括弧 ()
- CURRSYM 文字列
- TRIADSEP 文字
- RADIXSEP 文字
- 数字

入力文字列に CURRSYM 文字列が含まれている場合、関数はその文字列を AMOUNT に変換し、成功した場合には AMOUNT 型値を戻します。

入力文字列に RADIXSEP 文字が含まれている場合、関数はその文字列を FLOAT に変換し、成功した場合には FLOAT 型値を返します。

関数が文字列を NUMERIC に変換することに成功した場合には、NUMERIC 型値が返されません。

入力引数が正常に変換されなかった場合、"Bad arguments to a system function" というエラーメッセージが表示され、関数は NUMERIC 型のゼロ値を返します。

入力引数が NULL や空の文字列値であった場合、関数は NUMERIC 型の NULL 値を返します。