



リリース ガイド

Release Guide

| | |
|------------------------------------|------|
| サポートへのお問い合わせ | iv |
| ドキュメント | viii |
| オンライン ドキュメント | viii |
| 製本版マニュアル | viii |
| マニュアルの種類 | viii |
| 表記規則 | x |
| 特記事項 | xii |
| 1. 互換性について | 1 |
| 言語の互換性 | 2 |
| ITU SDL との互換性 | 2 |
| ITU MSC との互換性 | 5 |
| SDL の制限事項 | 7 |
| UML と MSC の制限事項 | 9 |
| TTCN の制限事項 | 11 |
| ツールの制限事項 | 12 |
| SDL Suite and TTCN Suite の制限事項 | 12 |
| SDL ツールの制限事項 | 16 |
| TTCN Suite ツールの制限事項 | 38 |
| 保存形式の互換性 | 45 |
| 既知の問題 | 50 |
| SDL and TTCN Suite ツール共通の既知の問題点 | 50 |
| SDL Suite の既知の問題 | 53 |
| TTCN Suite の既知の問題 | 59 |
| 2. リリース ノート | 63 |
| サポートされているコンポーネント | 64 |
| SDL Suite と TTCN Suite で共有するツールと機能 | 64 |
| SDL Suite ツールと機能 | 64 |
| TTCN Suite ツールと機能 | 65 |
| ユーザー用ドキュメント | 65 |
| 新しい機能と変更された機能 | 66 |
| SDL Suite and TTCN Suite ツールの共有機能 | 66 |



IBM Rational SDL and TTCN Suite 6.3
Release Guide

日本語版

サポートへのお問い合わせ

IBM Rationalソフトウェア・サポートへの問い合わせ

お手持ちのリソースで、問題が解決されない場合は、IBM® Rational® ソフトウェア・サポートに連絡してください。IBM ソフトウェア・サポートでは、製品の問題解決に関する支援を行っています。

前提条件

IBM Rational ソフトウェア・サポートに問題を送信するには、有効な Passport Advantage® ソフトウェア保守契約が必要です。パスポート・アドバンテージは、IBM の包括的ソフトウェア・ライセンスおよびソフトウェア保守 (製品のアップグレードおよび技術支援) オffering です。次のサイトからオンラインでパスポート・アドバンテージに登録できます。<http://www.ibm.com/software/lotus/passportadvantage/howtoenroll.html>

- パスポート・アドバンテージについて詳しくは、パスポート・アドバンテージ FAQ (http://www.ibm.com/software/lotus/passportadvantage/brochures_faqs_quickguides.html) にアクセスしてください。
- さらに支援が必要な場合は、IBM 担当員に連絡してください。

問題をオンラインで (IBM Web サイトから) IBM Rational ソフトウェア・サポートに送信するには、さらに以下が必要です。

- IBM Support Web サイトの登録ユーザーであること。登録について詳しくは、<http://www.ibm.com/software/support/> を参照してください。
- 許可された呼び出し元としてサービス要求ツールにリストされていること。

問題報告について

次のようにして、IBM Rational ソフトウェア・サポートに問題を送信します。

1. お客様の問題のビジネス・インパクトを判別します。IBM へ問題を報告する際は、重大度レベルを問われます。そのため、報告する問題とそのビジネス・インパクトを理解して、評価する必要があります。

重大度のレベルを決めるにあたっては、下表を参照してください。

| 重大度 | 説明 |
|-----|---|
| 1 | 問題は 危機的な ビジネス・インパクトを持ちます。プログラムを使用できず、業務に重大な影響が出ています。この状況には、即時に解決策が必要とされます。 |
| 2 | 問題は、 重大な ビジネス・インパクトを持ちます。プログラムは使用可能ですが、非常に限定されています。 |
| 3 | 問題は 部分的な ビジネス・インパクトを持ちます。プログラムは使用可能ですが、比較的重要でない(業務に大きな影響はない)機能が利用できません。 |
| 4 | 問題は わずかな ビジネス・インパクトを持ちます。問題による業務への影響がほとんどないか、問題に対する有効な回避策が実施済みです。 |

2. 問題を説明して、背景情報を収集します。IBM に問題を説明する際は、なるべく具体的に説明してください。IBM Rational ソフトウェア・サポートの専門家が、問題を解決するために効果的な支援をできるように、関連するすべての背景情報を含めてください。時間を節約するために、以下の質問の答えを用意してください。
 - 問題の発生時に実行していたソフトウェア（複数可）のバージョンは何ですか？

次のオプションを使用して、正確な製品名とバージョンを判別することができます。

- IBM Installation Manager を始動して、「ファイル」> 「インストール済みパッケージの表示」を選択します。パッケージ・グループを展開し、パッケージを選択して、パッケージ名およびバージョン番号を確認します。
- 製品を始動して、「ヘルプ」> 「製品情報」をクリックし、オフリング名とバージョン番号を確認します。

-
- オペレーティング・システムおよびバージョン番号(サービス・パックまたはパッチを含む)は何ですか?
 - 問題の症状に関連するログ、トレース、およびメッセージはありますか?
 - 問題を再現できますか?再現できる場合は、問題を再現するための手順は何ですか?
 - システムに変更を加えましたか?例えば、ハードウェア、オペレーティング・システム、ネットワーキング・ソフトウェア、またはその他のシステム・コンポーネントに変更を加えましたか?
 - 現在、問題に対する何らかの回避策を使用していますか?使用している場合は、問題の報告時にその回避策も説明する準備をお願いします。
3. IBM Rational ソフトウェア・サポートに問題を送信します。次の方法で、IBM Rational ソフトウェア・サポートに問題の送信ができます。

- **オンラインの場合:** IBM Rational ソフトウェア・サポートの Web サイト (<https://www.ibm.com/software/rational/support/>) にアクセスして、Rational サポート・タスク・ナビゲーターで「**サービス要求を開く (Open Service Request)**」をクリックします。エレクトロニック問題報告ツールを選択し、「問題管理レコード (PMR) (Problem Management Record (PMR))」を開き、問題についてご自身の言葉で正確に記述してください。

サービス要求を開く方法について詳しくは、

<http://www.ibm.com/software/support/help.html> にアクセスしてください。

IBM Support Assistant を使用してオンラインのサービス要求を開くこともできます。詳しくは、<http://www.ibm.com/software/support/isa/faq.html> を参照してください。

- **電話の場合:** 国または地域別の電話番号を調べるには、<http://www.ibm.com/planetwide/> の「IBM directory of worldwide contacts」で、お住まいの国名または地域名をクリックします。
- **IBM 担当員に依頼する場合:** オンラインまたは電話で IBM Rational ソフトウェア・サポートにアクセスできない場合は、IBM 担当員に連絡してください。必要な場合は、お客さまに代わって、IBM 担当員がサービス要求を開くことができます。<http://www.ibm.com/planetwide/> で、各国への詳しい連絡先情報を検索できます。

送信した問題が、ソフトウェアの障害に関するものか、資料の欠落や不正確な記述によるものである場合は、IBM Rational ソフトウェア・サポートはプログラム診断依頼書 (APAR) を作成します。APAR には、問題の詳細が記述されます。IBM ソフトウェア・サポートは可能な限り、APAR が解決されてフィックスが提供され

るまでの間に実施できる回避策を提供します。IBM は、同一の問題を経験している他のユーザーが同じ解決方法を利用できるように、IBM Rational ソフトウェア・サポート Web サイトに解決済みの APAR を公開し、毎日更新しています。

ドキュメント

SDL Suite and TTCN Suite製品のすべてのユーザー マニュアルは、オンラインで入手できます。その一部は、印刷されたマニュアルとして提供されています。

オンライン ドキュメント

CD-ROMには、HTMLファイル形式オンライン ドキュメントが収められています。

コンテキスト対応のヘルプを使うことにより、SDL Suite and TTCN SuiteからHTMLファイルにアクセスすることができます。

製本版マニュアル

オンライン マニュアルはPDF形式でSDL Suite and TTCN SuiteのCD-ROMにおさめられています。マニュアルの一部を選択して、画面に表示したり印刷をすることができます。

マニュアルの種類

以下のマニュアルが用意されています。

- [Installation Guide](#) :

SDL Suite and TTCN Suiteのインストール方法について記載しています。また、コンピュータ環境の設定方法やライセンスの処理方法について説明しています。

- [Release Guide](#) (本マニュアル) :

新機能や、制限事項、互換情報など、製品の新しいバージョンに関する情報が記載されています。互換情報では、SDL Suite 6.3 and TTCN Suite 6.3と以前のバージョンとの互換性や、サポートされる言語と標準規格で規定されている言語との互換性について説明しています。

- [Methodology Guidelines](#) :

SDL Suiteを使ったソフトウェア開発におけるSDLやそのほかの表記方法の具体的なガイドラインを説明しています。

-
- [TTCN Suite 6.2 Methodology Guidelines](#) :

TTCN SuiteでTTCN言語を使う際に役立つ記述方法について記載していません。TTCNの言語体系や概念、データ型について説明しています。

- [Getting Started](#) :

SDL Suite、およびこれらの環境でサポートされる各種言語と表記方法についての入門書です。記載されている各チュートリアルを学習および実習することにより基本的なSDL Suite ツールの使用方法を習得することができます。

また、このマニュアルにはSOMT方式、Cmicroターゲティングの理解に役立つ説明があります。各説明には、実践的な演習の項目も加えられています。

- [TTCN Suite 6.2 Getting Started](#) :

TTCN Suiteを習得するための、TTCN、TTCN Suiteの入門とTTCN Suite ツールのチュートリアルが記載されています。

- [User's Manual](#) :

全ツールおよび機能に対する、リファレンス情報とガイドラインの資料が記載されています。このドキュメントには、すべてのウィンドウ、ダイアログボックス、メニュー、コマンド、ファイル、保存形式、オプションなどの情報が網羅されています。また、ツールの使用方法やさまざまなタスクの実行方法の手引きも説明されています。

表記規則

ここでは、すべてのマニュアルで使用している表記規則について説明します。

メニュー コマンドとユーザー インターフェイス オブジェクト

All ユーザー インターフェイスである、ラベル、メニュー名、メニュー項目、ボタン名には、以下のように[] (かど括弧) を使います。

[編集]メニューには[貼り付け]コマンドがあり...

[OK]ボタンをクリックして、...

また、メニューのコマンドは以下のように「-」(ハイフン) を使って、一連のメニュー、サブメニュー、メニュー項目の選択を単一表記する場合があります。

[編集]-[貼り付け]の順に選択し...

[ツール]-[リンク]-[リンク先へ移動]コマンドを...に使います。

ファイル名、変数、その他

ファイル名やディレクトリ名、変数名、ソース コード、ファイル キーワードなどは、以下のようにタイプライター フォントで表記します。

sdt.sslファイル...

\$telelogic環境変数を設定...

入出力テキスト、リスト、その他

ソース コードの抜粋、ファイルの内容、コンピュータから出力されたテキストなども、以下のようにタイプライター フォントで表記します。

```
State1:  
  Sig1: INPUT  
  Sig2: SAVED  
State2:  
  Sig3: INPUT  
  Sig1: UNHANDLED
```

操作の説明で、キーボードから入力しなければならない文字を示す場合は、以下のように「」(かぎ括弧) を使います。

「Convert, Demon, System」と入力します。

「C:¥Telelogic¥SDL_TTCN_Suite6.3J> mkdir example」

プレースホルダ

<> (不等号かっこ) によって表記された名前は、抽象名を表し、具体名で置き換える必要があります。表記例は以下のようになります。

この構文は、「<プロセスインスタンス> <変数名>」の形式で...

複数のキー

特殊なキーや複数のキーを同時に入力しなければならない場合は太字で表記します。複数のキーは、以下のように「+」（プラス記号）で連結表記します。

このコマンドを実行する際には、Returnキーを押し...

Ctrl+Cキーを押して

ショートカットキーは、**Ctrl+Meta+V**...

プラットフォーム間の相違点

マニュアルは、**UNIX**と**Windows**の双方のプラットフォームに対して共通です。プラットフォーム間で相違点がある場合は、文中にプラットフォーム専用であることを示す記述をします。プラットフォームを示す記述には、以下のように必ず、「**UNIX**」または「**Windows**」という名称を加えて表します。

この情報は、**UNIX**の場合は**.sdtpref**ファイルに、また、**Windows**の場合は**sdt.ini**ファイルに保存されます。

XYZ は以下の要素によって成り立っています。

- アルファ
- ベータ (**UNIX**のみ)
- ガンマ
- デルタ (**Windows**のみ)
- その他

項の記載内容がすべてプラットフォーム専用の場合は、以下のようにその項の見出しにプラットフォーム名を記述します。

操作の実行 (**UNIX**)

操作の実行 (**Windows**)

画面表示は、**UNIX**か**Windows**のいずれかの画面を掲載します。実際にスクリーンに表示される画面と挿入図との体裁やレイアウトが若干異なることがあります。が、機能的には同じです。

特記事項

本書は、**IBM Rational SDL Suite 6.3** および **IBM Rational TTCN Suite 6.3** および新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

© Copyright IBM Corporation 1993, 2009.

著作権表示

本書は米国 **IBM** が提供する製品およびサービスについて作成したものであり、本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 **IBM** の営業担当員にお尋ねください。本書で **IBM** 製品、プログラム、またはサービスに言及していても、その **IBM** 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、**IBM** の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、**IBM** 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

Copyright © 2009 by IBM Corporation.

IBM 特許権

IBM は、本書に記載されている内容に関して特許権（特許出願中のものを含む）を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-8711
東京都港区六本木 3-2-12
日本アイ・ビー・エム株式会社
法務・知的財産
知的財産権ライセンス渉外

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム（本プログラムを含む）との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、製造元に連絡してください。

Intellectual Property Dept. for Rational Software |
IBM Corporation
1 Rogers Street
Cambridge, Massachusetts 02142
U.S.A.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、**IBM** 所定のプログラム契約の契約条項、**IBM** プログラムのご使用条件、ま

たはそれと同等の条項に基づいて、IBM より提供されます。

保証の不適用

以下の保証は、国または地域の法律に沿わない場合は、適用されません。IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

機密情報

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

追加の法的通知が、本書で説明するライセンス付きプログラムに付随する「プロ

グラムのご使用条件」に含まれている場合があります。

サンプルコードの著作権

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、**IBM** に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って **IBM** は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

©(お客様の会社名) (西暦年). このコードの一部は、**IBM Corp.** のサンプル・プログラムから取られています。

IBM の商標

IBM および関連の商標については、www.ibm.com/legal/copytrade.html をご覧ください。これは、**IBM** が現在所有する米国における商標の最新リストです。以下は、**International Business Machines Corporation** の米国およびその他の国における商標です。

このページには、**IBM** が使用しているすべてのコモン・ロー商標は掲載されていません。**IBM** が販売している製品は多数あるため、コモン・ロー商標のうち、最も重要な商標のみを掲載しております。このページに商標が掲載されていなくても、それは **IBM** がその商標を使用していないということではなく、その製品が現在販売されていない、または関連する市場で、その製品が重要ではないということを示すものではありません。

他社の商標

Adobe、**Adobe** ロゴ、**PostScript** は、**Adobe Systems Incorporated** の米国およびその他の国における登録商標または商標です。

Java およびすべての **Java** 関連の商標およびロゴは、**Sun Microsystems, Inc.** の米国およびその他の国における商標です。

Linux は、**Linus Torvalds** の米国およびその他の国における商標です。

Microsoft、**Windows**、**Windows 2003**、**Windows XP**、**Windows Vista** および/またはその他の **Microsoft** 製品は、**Microsoft Corporation** の米国およびその他の国における商標または登録商標です。

Pentium は、**Intel Corporation** の商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。



互換性について

本章では、**SDL Suite** と **TTCN Suite** の旧リリースと **SDL Suite 6.3** と **TTCN Suite 6.3** との互換性、およびバージョン **2.3** と **3.X** との互換性について説明します。また、**SDL Suite** でサポートする言語の互換性についても説明します。

ここでは、保存形式と言語サポートを中心に話を進めます。**TTCN Suite** に関しては、古い **TTCN Suite** データベースをアップグレードする方法と、以前のバージョンの **TTCN Suite** で **TTCN Suite** データベースを開く方法を説明します。

プラットフォームおよび製品の互換性の詳細については、[第1章「プラットフォームおよび製品」5ページの「互換性マトリックス」](#) および [6ページの「構築および動作保証マトリックス」](#) を参照してください。

言語の互換性

ITU SDLとの互換性

SDTのSDL言語サポートには、ITU-T Z.100勧告で定義されているSDL'92のほとんどすべてが含まれます。

SDLでサポートされるSDLとZ.100勧告との主な相違点は以下のとおりです。

- ジェネリックなシステム概念が部分的にしかサポートされません。
- マクロ呼び出しはフローチャートダイアグラム内でのみ可能です。
- ダイアグラムの名前とダイアグラム ページの名前はSDLの規則に準拠しますが、スペースは使用できません。

制限事項の詳細については、[第1章「互換性について」7ページの「SDLの制限事項」](#)および[18ページの「SDLエディタ」](#)を参照してください。

SDLエディタは、SDL/GRの表記法に対し、以下の拡張機能をサポートします。

- *追加ヘッダーシンボル*（継承、特殊化、仮パラメータ、信号集合などの定義に使用します）。
- *演算子参照シンボル*（オーガナイザで処理されるSDL構造で演算子を表示します）。

SDL Suiteでは、いくつかの新しいSDL概念をサポートしています。中には、SDL-96として知られる最新のZ.100勧告の補遺で定義されているものもあります。また、SDLに対するIBM Rational独自の拡張機能も含まれています。拡張機能は、すべての関連するツールでサポートされます。

SDL-96拡張機能

- *チャンネルおよび信号経路のリモートプロシージャ*：この拡張機能によって、チャンネルの信号リストおよび信号経路にリモートプロシージャの名前をつけることができます。この拡張機能の主な利点は、SDLシステムの環境に実装されているリモートプロシージャをプロセスが呼び出せるようになることと、逆に、SDLシステムの環境からリモートプロシージャを呼び出せることです。
- *外部プロシージャ*：外部プロシージャは標準のSDLプロシージャのように呼び出されますが、ターゲット言語（C）に実装されます。これによって、Cの関数をSDLで簡単に使用できるようになります。

- 外部演算子：外部演算子は標準のSDL演算子のよう呼び出されますが、ターゲット言語（C）に実装されます。これによって、Cの関数をSDLで簡単に使用できるようになります。

上記以外のSDL-96の機能はSDL Suiteではサポートされません。

SDL Suite 独自の拡張機能

以下は、SDL Suite独自のSDLの拡張機能です。これらの拡張機能の詳細については、『[Methodology Guidelines](#)』の第2章、「データ型」および第3章「[SDLの拡張表現の使用](#)」を参照してください。

- **Choice**：この構文を使用すると、値にいくつかのコンポーネントの中の1つのみを与えることができるデータ型を定義することができます。Choice型は、#UNIONコードジェネレータディレクティブの代わりになります（ただし、#UNIONは下位互換性があるため今後もサポートされます）。
- 構造体のオプション要素とデフォルト要素：これで、構造体型のフィールドをオプションにしたりデフォルト値に指定できます。この機能は、プロトコルの設計および実装に役立ちます。
- サイズの制約：この構文を使用すると、長さに制限のある型を定義することができます。
- ビットフィールド：構造体型の整数フィールドを、Cと同様のビットフィールドとして指定することができます。この機能は、SDLにビットフィールドを含むCのデータ型を再利用するときに役立ちます。

- **Z.105型およびジェネレータ** : Z.105では、以下に示すような定義済のデータ型およびジェネレータが新たに追加されました。**Bit**、**Bit_String**、**Octet**、**Octet_String**、**Object_Identifier**、**IA5String**、**NumericString**、**PrintableString**、**VisibleString**、**Null**、および**Bag generator**。
- **Powerset演算子** : **Powerset**が**Length**演算子および**Take**演算子によって拡張されました。これにより**Powerset**を繰り返すことが可能になるため、**Powerset**がより便利になります。
- **ポインタ** : **SDL**でポインタを操作しやすくする構文拡張機能を用意しました。ポインタを使用するためには、**ctypes** パッケージの**Ref**ジェネレータを使用する必要があります。
- **Own**および**Oref** : **Own**ジェネレータおよび**ORef**ジェネレータは、セーフポインタをサポートします。このポインタを使用すると、信号の送信中にデータをコピーする代わりにポインタ値を送信することにより、アプリケーションの処理速度を上げることができます。メモリの割り当ておよび割り当て解除は自動的に行われます。
- **SDLのアルゴリズム** : 新しい構文が多数追加されて、アルゴリズムがタスクシンボル内に表示されるようになりました (たとえば、**if-then-else**、ループ、分岐など)。
- **拡張演算子** : パラメータがない演算子、結果のない演算子、演算子への**in/out**パラメータ (演算子シグネチャとダイアグラムの両方) のサポートが追加されました。代入文の代替えとして、演算子アプリケーション文は、値を返さない演算子を起動する場合があります。
- 外部変数がサポートされました。
- **Latin One** : **ISO-Latin-1**が定義済のソート文字でサポートされました。
- **I18n** : 定義済のソート**charsting**は、マルチバイトの文字リテラルをサポートします。
- 命名規則 : アンダースコアは、制限なく名前に使用できます。
- **Make**演算子が定義済みのアイテム**String**、**Powerset**、**Bag**、および**Object_identifier**に加えられました。この演算子は0個以上の引数を取り、**"(.)"**の表記により起動します。**Make**演算子のシグネチャは以下のとおりです。

```
make! : * Itemsort -> String;
make! : * Itemsort -> Powerset;
make! : * Itemsort -> Bag;
make! : * Natural -> Object_identifier;
```

*は変数の引数 (0以上の値が認められている) を表します。

- 定義済みのデータ型 `Object_identifier` の新しい演算子は以下のとおりです。

```
append      : in/out Object_identifier, Natural;
```

SDL-2000拡張機能

- `Integer`、`Bit_string`、`Octet`、`Octet_string` ソートのバイナリ リテラルと16進リテラル。例: `"b '1001'B 'H 'Fe'h`
- ブロックとプロセスの混在
 - － システムには、ブロックを介さずにプロセスを配置できます。
 - － ブロックには、サブストラクチャを介さずにブロックを配置できます。
 - － チャンネルは、プロセスに接続できます。

ITU MSCとの互換性

SDLのMSC言語サポートでは、基本MSC（MSC'93）を完全にサポートし、ITU-T Z.120勧告で定義されているMSC'96も部分的にサポートしています。

MSC'96サポートに関しては、以下の点が重要になります。

- ロードマップ、つまりハイレベルMSC（HMSC）が、HMSCエディタでサポートされます。ただし、コメントシンボルおよびパラレルフレームシンボルは除きます。
- MSCダイアグラムは、MSC参照式によって参照シンボルをサポートします。このため、ほかの（H）MSCダイアグラムの参照が可能になります。
- MSCダイアグラムは限定子をサポートします。
- MSCダイアグラムはインライン表記をサポートします。これにより、複数のシナリオを同じダイアグラムで表現することができます。MSCエディタはゲートをサポートしないので、インライン表記は、すべてのインスタンスを常にカバーします。
- MSCダイアグラムでは独立したタイムシンボルがサポートされます。
- MSCダイアグラムは、失われたメッセージおよび見つかったメッセージ（ブラックホールおよびホワイトホール）をサポートします。

MSCサポートの制限事項については、[第1章「互換性について」9ページの「UMLとMSCの制限事項」](#) および [16ページの「MSCエディタ」](#) を参照してください。

MSCエディタでは、Z.120の拡張機能もいくつかサポートしています。

- Z.100勧告をマップするステータスに、メッセージおよびタイマを追加で割り当てることができます。これにより、図式トレースをより直観的に表現することが可能になります。
- 出力イベントと入力イベントが同じインスタンスに存在する場合は、メッセージを *self* に送信することができます (Z.120では、これを明示的には許可も禁止もしていません)。

MSC'96の予約語

MSC'96では、以下の単語が予約されています。

action, all, alt, as, before, begin, block, by, comment, concurrent, condition, connect, create, decomposed, empty, end, endconcurrent, endmsc, endexpr, env, exc, expr, external, found, from, gate, in, inf, inline, inst, instance, loop, lost, msc, mscdocument, msg, opt, order, par, process, reference, related, reset, service, seq, set, shared, stop, subst, system, text, timeout, tim, to, via

対応する言語の構文をMSCエディタがサポートしていない場合でも、予約語を名前や識別子として使用することはできません。これは、エディタがMSC'96言語を現時点以上にサポートする場合や、異なるサブセットの言語をサポートするほかのツールにMSC/PRをエクスポートする場合に、互換性の問題が発生するのを防ぐためです。

MSC'93の予約語

MSCエディタおよびHMSCエディタでは以下の単語も予約語として扱います。ただし、これに対応するMSC'96の予約語にマップされます。

| MCS'93 | MSC'96 |
|--------------|----------|
| instancehead | instance |

このマッピングは、古いスクリプトやMSC/PRジェネレータを動作させるために行われます。

サポートされない予約語

MSCエディタおよびHMSCエディタでは以下の予約語をサポートしません (つまり、名前や識別子に関しては予約語として扱われますが、エディタでは、これらの単語を使用して作成する構文をサポートしません)。

before, begin, external, found, gate, lost, order, related, via

また、エクスプローラは以下の予約語をサポートしません。

subst, by, msg

パラメータ リスト

MSC/PRインポートでは、下位互換性のため、タイマの設定時にパラメータリストとタイマー時間名を空白にすることができます。これは、MSC'93とMSC'96のいずれにも従っていません。

SDLの制限事項

ここでは、1992年のITU-T Z.100勧告に関連して、SDLの主な制限事項について説明します。この制限は、1つまたは複数の特定のツールに対して存在します。したがって、1つの概念が完全にはサポートされていないなくても、ツールセットの限られた部分では使用可能な場合があります。

ここでは、以下のツールの制限事項について説明します。

- SDLエディタ
- GR-PRコンバータ
- SDLアナライザ
- SDL to Cコンパイラ (CbasicおよびCadvanced)

アナライザおよびCコンパイラにおけるSDLの制限事項の詳細については、[20ページの「SDLアナライザ」](#) and [23ページの「CAdvanced SDL to Cコンパイラ」](#)を参照してください。

シミュレータ、エクスプローラ、TTCNリンクカーネル、およびCmicro SDL to Cコンパイラに関するSDLの制限事項については、『User's Manual』の以下のセクションを参照してください。

- [第49章「SDLシミュレータ」](#) [2206ページの「制限事項」](#)
- [第52章「The SDL Explorer」](#) [2362ページの「Restrictions」](#)
- [第35章「TTCN Test Suite Generation」](#) [1405ページの「SDL Restrictions」](#)
- [第65章「The Cmicro SDL to C Compiler」](#) [3425ページの「Restrictions in Combination with SDL Target Tester」](#)

| Z.100の章 | 概念と制限事項 | 影響を受けるツール |
|--|--|-----------------------------|
| 2.2.2 Visibility rules, names and identifiers | <name>:<name>内のスペースはサポートされません(スペースは<underline>の簡略記法です)。 | SDLエディタ、アナライザ、SDL to Cコンパイラ |
| 2.4.1.2 Package | <interface>定義、およびuseステートメント内の<definition selection list>はサポートされません。 | アナライザ、SDL to Cコンパイラ |
| 2.5.1 Channel | nodelay :delay付きチャンネルとnodelay付きチャンネルは同じに扱われます。 | SDL to Cコンパイラ |
| 2.7.4 Output | <block instance set>に対する via all は単一の信号のみを送ります。 | SDL to Cコンパイラ |
| 2.9 Internal input and output | <internal input symbol>および<internal output symbol>は使用できません。 | SDLエディタ |
| 3.2 Partitioning | <channel substructure definition>は、生成したコードではサポートされません。 | SDL to Cコンパイラ |
| | プロセス定義とブロック定義の両方を持つサブストラクチャでは、プロセス定義が無視されます。 | アナライザ、SDL to Cコンパイラ |
| 3.3 Refinement | <signal refinement>はサポートされていません。 | SDL to Cコンパイラ |
| 4.2 Macro | <macro definition>:マクロとマクロ呼び出しはPRに変換され、PRフォームで展開されます。これによりマクロの使い勝手が制限されます。 | アナライザ |
| | <any area>:図式マクロには、シングルフローのフローシンボルのみを含めることができます。 | SDLエディタ、GR to PRコンバータ |
| | <macro call>:図式マクロ呼び出しは、プロセスより上位のスコープでは使用できません。 | SDLエディタ、GR to PRコンバータ |
| 4.3.3 Optional definition | <select definition>:テキスト形式で定義できるようになりました(グラフィック形式には未対応)。選択式に使われる(外部)シノニムは、システム定義の中で定義しなければなりません、タイプベースとは限りません。 | SDLエディタ、アナライザ、SDL to Cコンパイラ |

| Z.100の章 | 概念と制限事項 | 影響を受けるツール |
|---|--|---------------------|
| 4.3.4 Optional transition string | <alternative question>: 遷移オプションの質問部に外部シノニムを含めることはできません。 | アナライザ、SDL to Cコンパイラ |
| 5.2.3 Axioms | <axioms>: サポートされません。 | アナライザ、SDL to Cコンパイラ |
| 5.2.4 Conditional equations | <conditional equation>: サポートされません。 | アナライザ、SDL to Cコンパイラ |
| 5.3.1.14 Name class literals | <name class literal>: サポートされません。 | アナライザ、SDL to Cコンパイラ |
| 5.3.1.15 Literal mapping | <literal mapping>: サポートされません。 | アナライザ、SDL to Cコンパイラ |
| 5.4.4.5 Timer active expression | <timer active expression>: パラメータのあるタイマで使用できません。 | SDL to Cコンパイラ |
| 6.2 Context parameter | <formal context parameters>および<actual context parameters>はサポートされません。 | アナライザ、SDL to Cコンパイラ |

UMLとMSCの制限事項

ここでは、UMLに対するOMG仕様や、MSCに対するITU-T勧告に関連する、SDLでのUMLとMSCの主な制限事項について説明します。

以下の説明に記述されている参照は、OMGドキュメント「ad/99-06-08」(UML Notation Guide, version 1.3, June 1999)とZ.120勧告の各項を指しています。

UMLクラスダイアグラム

UML 1.3静的構造(クラス)ダイアグラムの以下の機能は、サポートされません。

- パッケージ (3.13項)、ただし、クラス名でのパッケージ参照は可能です。
- インターフェイス (3.28項)
- パラメータで表されたクラス、またはテンプレート (3.29項)
- 複合オブジェクト (3.39項)
- Xor関連 (3.41.5項)
- 関連の操作性 (3.42.2項)

- N-ary関連 (3.46項)
- 一般化の制約条件 (3.49.2項)
- 依存性、すなわち点線の矢印 (3.50項)

UMLステートチャートダイアグラム

UML 1.3ステートチャートダイアグラムの以下の機能はサポートされません。

- 背後の要素を表わすアイコン (3.76項)
- 同時状態によるサブ状態のフィールド (3.76項)
- 遷移時間 (3.78項)
- 同時遷移 (3.79項)
- 履歴状態インジケータ (3.80項)
- スタブ遷移 (3.80項)
- 遷移経路の確率化 (3.81項)
- Submachine状態 (3.82項)
- Synchron状態 (3.83項)

UML実装ダイアグラム

DPエディタダイアグラムで使用する表記規則は、UML 1.3の実装ダイアグラムをベースにしています。DPエディタが実装ダイアグラムと異なる主な点を以下に示します。

- (ノードや要素に対して) 点線矢印による依存性は使用できません。
- (要素に対する) インターフェイスは使用できません。
- (ノード間の要素や要素間のオブジェクトの) 移行は使用できません。
- 内容を示す図式の入れ子構造は (複合アグリゲーションに対してのみ) 使用できません。
- スレッドの概念が加えられています。

MSCとHMSC

- イベントの全順序付けはサポートされません (Z.120第4.5章)。
- ゲートはサポートされません (Z.120第4.4章)。
- HMSCダイアグラムのコメントシンボルはサポートされません。
- HMSCダイアグラムの並列フレームはサポートされません (Z.120第5.5章)。

TTCNの制限事項

- TTCNでサポートされるTTCN言語は、「Delivery 8.3, August 2, 1996」というラベルの付いた、新しいTTCN標準の暫定バージョンに準じていますが、モジュールのTTCN機能のみDelivery 9.6バージョンに準じています。
- Delivery 9.6モジュールのTTCN機能のために、TTCNパッケージのドキュメントタイプをサポートは行われていません。
- モジュールのTTCN機能のインポート/エクスポートメカニズムは、現時点では、名前付き番号のインポートに必要な「xxx[yyy]」構文を使用するインポートを処理できません。また、名前が衝突しているオブジェクトをインポートするための「xxx::yyy」構文も処理できません。

TTCNでサポートされるTTCNとASN.1の詳細な制限事項と変更点については、[『User's Manual』の第38章、「Languages Supported in the TTCN Suite」](#)を参照してください。

ツールの制限事項

SDL Suite and TTCN Suiteの制限事項

オペレーティングシステムとウィンドウ環境

- X端末またはXサーバー ソフトウェアを搭載しているパーソナルコンピュータでSDL Suite and TTCN Suiteを実行する場合、マウスのクリックが、SDL Suite and TTCN Suiteの認識できる速度で伝わらないことがあります。

UNIX端末の環境変数DISPLAYがローカルのXサーバーに設定されている場合は、Xリソースの*multiClickTimeを使用して、デフォルトのしきい値の250 msを変更することができます。

```
xrdb -merge
*multiClickTime: 1000
^D
```

適切な値（ミリ秒）は、使用するハードウェアとネットワークによって異なります。

オンライン ヘルプとドキュメント

- オンラインヘルプの索引からリンクを移動したときには、HTMLヘルプファイル内の正確な位置に表示されない場合があります。場合によっては、参照した説明文の数行下に移動する場合があります。また、表の場合は、表よりも後に移動することがあります。

これを回避するには、対応するトピックについて記述しているヘッダ、記号付き項目、あるいは表の行が表示されるまでテキストを上方向にスクロールします。

- 『User's Manual』の第78章、「[SOMTチュートリアル](#)」では、リンクマネージャの基本的な知識を持っていることが前提になっています。ただし、このツールはこれより前のチュートリアルの章で説明されていません。リンクマネージャの詳細については、『User's Manual』の第9章「[Implinks and Endpoints](#)」の419ページ、「[Implinks and Endpoints](#)」を参照してください。
- ステートチャートエディタおよびHMSCエディタに関するチュートリアルはありません。
- MSCエディタではMSCのいくつかのシンボルとラインの外観が更新されていますが、ドキュメントのすべてのMSCダイアグラムの記述は、これに伴って更新されていません。

- ドキュメントに含まれているすべてのダイアログボックスやダイアグラムでは、パスなどの記述に前バージョンの名称がそのまま残り、SDL Suite 6.3に変更されていません。
- Netscapeの最後のインスタンスがブラウザではなくアプリケーションであった場合、ヘルプファイルは開きません。Help*NetscapeCommandの設定をnetscape -noraize -remote 'openURL(URLTOOPEN,new-window)''にすることで、この問題を解決できます。

ASN.1ユーティリティ

ASN.1ユーティリティは、拡張機能X.681およびX.682を含むITU-T勧告X.680で定義されているすべての（ASN.1の）構文を処理します。X.683構文の一部もサポートされています。サポートされていない構文については、以下の制限をご覧ください。古いASN.1バージョンX.208で定義されている機能はサポートされません。

X.680に対する制限事項

ASN.1ユーティリティでは、X.680で定義されているすべての構文を処理可能ですが、以下のX.680の意味論の概念はサポートされません。

- モジュールはモジュール名によってのみ区別されます。**GlobalModuleReference**では、モジュール名に続くオブジェクト識別子は無視されます。この場合、モジュール名は、常に限定子と見なされます。
- EmbeddedPDVType**、**ExternalType**、**UnrestrictedCharacterStringType**には値を割り当てることができません。これらのデータ型を制約することはできません。
- GeneralizedTime**と**UniversalTime**データ型を表現する文字列の内容はチェックされません。

X.681に対する制限事項

- オブジェクトクラス参照（X.681, 7.1）に小文字を使用できます。オブジェクトクラス参照は、タイプ参照で指定された文字列で構成されます。
- 小文字と数字をワード項目に含めることができます（X.681, 7.9）。ワードは、タイプ参照で指定された文字列で構成されます。

X.682に対する制限事項

- コンポーネント関係がある型に値を代入する場合、Atマークはチェックされません。

- 1つの型に複数のテーブル制限を行うことができません。たとえば、"**MY-CLASS.&TypeField** (**{MY-SET1}**) (**{MY-SET2}**)"は許可されません。

X.683に対する制限事項

- パラメータで表示されたオブジェクトクラス、オブジェクトおよびオブジェクトセット、値セットデータ型はサポートされません。上記のエンティティはすべてパラメータとしては使用できますが、指定できるのはパラメータで表示されたデータ型と値だけです。
- ダミーのガバナーはサポートされません。

X.690およびX.691に対する制限事項

- **ASN.1** コーダでは、コンポーネントを識別するための参照は1つしかサポートされません。**AtNotationList** (**X.681, 10.7**参照) で複数のコンポーネントの関係により制限されているコンポーネントについては、最初の制限のみが**ASN.1** コーダによって処理されます。
- **X.690 8.11.3**の制限。デフォルト設定でコンパイルした場合、データ値の順序は**X.680 8.4**に規定されている標準的な順序に制限されます。これを回避するには、**CODER_BER_CANONICAL_OFF**コンパイルスイッチを指定する必要があります。
- **X.680**で定義されている以下の**ASN.1**型は、**ASN.1**エンコーダおよびデコーダではサポートされていません。
 - **BMPString**
 - **GeneralString**
 - **GraphicString**
 - **ISO646String**
 - **TeletexString**
 - **T61String**
 - **UniversalString**
 - **UTF8String**
 - **VideotexString**
 - **UnrestrictedCharacterStringType**

- EmbeddedPDVType
- ExternalType
- InstanceOfType
- ObjectDescriptor

Z.105に対する制限事項

Z.105の以下の概念は、ASN.1ユーティリティではサポートされません。

- **ASN.1定義をSDLダイアグラムのテキストシンボルに置くことはできません。**ASN.1定義は、独立したASN.1モジュールにのみ置くことができます。一方、Z.105では、SDLとASN.1の定義を自由に組み合わせて使用することができます。
- 別のモジュールで定義されたデータ型の値表記はサポートされません。たとえば、`val X ::= 3`は、Xが同じモジュールで定義されている場合のみ使用できます。したがって、代わりに外部の値参照を使用する必要があります。
- **bit string、octet string、set、set of、sequence、およびsequence of**については、SDL式内でASN.1値表記を使用できません。
- **Make**演算子におけるオプションのSDLサポートが有効になっていない場合、オプションまたはデフォルトのコンポーネントが値を代入しているSEQUENCE/SETの値表記を、SDLにマップすることはできません。
- **UniversalString、GraphicString、およびEXTERNAL**は同じ名前のSDLデータ型に変換されますが、これらのデータ型に対応する実装はありません。これは、IBM RationalがZ.105でのマッピングを不十分と考えているためです（拡張文字セットを実際に使用することはできません）。
- **REAL**型の使用には制限事項がいくつかあります。**REAL**値は制約で使用できますが、**REAL**値を値範囲で使用することはできません。以下に例を示します。

```

SDL_REAL1 ::= REAL ( 0 ) -- OK
SDL_REAL2 ::= REAL ( { mantissa 2, base 10, exponent 1 } ) -- OK
SDL_REAL3 ::= REAL ( 0 .. PLUS-INFINITY ) -- FAIL
SDL_REAL4 ::= REAL ( MINUS-INFINITY .. v4 ) -- FAIL

```

- **PLUS-INFINITY**および**MINUS-INFINITY**は、大きな数に変換されます。使用するプラットフォームやアプリケーションに対して、この数が十分大きくない場合は、これらのSDLシノニムをユーザーが定義する必要があります。

- 名前の競合が、名前付きビット、名前付き番号、またはASN.1値で発生している場合には、最初に出現するシノニムがSDLに生成されます。たとえば、ASN.1定義が、`T1 ::= BIT STRING { a(3) }, b INTEGER ::= 11,` および `T2 ::= INTEGER { a(4), b(5) }` の場合、`a`がBIT STRINGで`b`がINTEGERのシノニムだけが警告メッセージと共にSDLに対して生成されます。これは、Z.105標準に従ってすべてのシノニムを生成すると、SDLアナライザエラーが発生するためです。
- SDL(アナライザ)のリテラルとシノニムに同じ名前がある場合、競合が発生します。ASN.1定義が `T ::= ENUMERATED {x, y} x INTEGER ::= 15 val T ::= x generated SDL newtype T literals x, y operators ordering; endnewtype; synonym x Integer = 15; synonym val T = x;` の場合コンパイルは実行されず、「エラー 375 シノニムからリテラルのデータ型が一致していません:synonym val T=?x」と「情報 352 xは可視定義の1つです」というメッセージが通知されます。ただし、この問題を解決する方法は簡単で、生成されたSDLでxの値の前に「`synonym val T = <<type T>>x;`」のように限定子を付けるだけです。

印刷

- 1ページに1ファイルが対応するEPSファイルを印刷するときには、ページ名に従ってファイル名が付けられ、その際大文字と小文字が区別されます。Windowsの場合に、複数のページに同じ名前が付けられ、さらに大文字と小文字でのみ区別可能なときには、最終ページを含んだファイルのみが印刷されます。この処理は、Windowsで大文字と小文字が区別されない制限が考慮されたものです。
- FrameMakerやInterleafで印刷する場合、カラー印刷はできません。

SDLツールの制限事項

すべてのエディタ

- ダイアグラム名は最大255文字です。

MSCエディタ

- インライン表記は常にすべてのインスタンス軸をカバーします。この制限は、ゲート（特にインラインゲートインターフェイス）がサポートされないことによるものです。

- インスタンスの列フォームと共通領域はサポートされません。行フォームのみサポートされます。
- SDLページ名に、"pg__name"のように連続する2つのアンダースコアが含まれている場合、MSCにログ出力するときに構文エラーが発生します。「MSCサービスINSERTOBJECTがエラーになりました。」で始まるエラーメッセージが通知されます。

OMエディタ

- シンボルに構文エラーが含まれる場合は、[クラスを表示および編集]ダイアログボックスの使用は慎重に行ってください。

属性と操作に対する構文チェッカーにはエラー回復機能があり、必要のない回復処理を行うことがあります。構文エラーが検出されたときに[クラスを表示および編集]ダイアログボックスを使用して**編集を行う**場合は、細心の注意が必要です。描画領域では、構文エラーは、シンボルの対応するテキストフィールドに赤い下線で強調表示されます。ただし、[クラスを表示および編集]ダイアログボックスの参照機能は、構文エラーが検出された場合でも考慮せずに使用できます。

UML2SDL ユーティリティ

- 現時点では、UMLダイアグラムにおける定義済みのSDL型の使用を特に認めていません。そのため、UML SuiteのPid型に属性を持たせると、Pid型について、(誤りのある)新しい型定義が生成されます。

テキストエディタ

- 多くのエンドポイントを含む広い領域のテキストを編集する場合、テキストエディタでの更新が遅くなる可能性があります。

パフォーマンスに問題がある場合は、対策として、エンドポイントの表示をオフにしてください。

- Windowsの場合、単語をダブルクリックして選択し、別の語句を入力して単語を置き換えると、ウィンドウに表示されるテキストとエディタ内部で処理される実際のテキストとの間に不一致が発生します。新しい単語の後ろに表示される空白文字は、内部的には認識されません。そのため、テキストを保存し、後で開いたときにはなくなっています。選択した何らかのテキストが改行文字で終わっている場合も同様です。選択した文字を新しいテキストで置き換えた場合、そのテキストを保存する際に末尾の改行文字は削除されます。

英語版Windowsが稼動する環境では、中国語は使用できません。英語版Windowsが稼動する環境で中国語のサポートを利用している場合は、テキストエディタで中国語を入力できます。ただし、テキストファイルを開き直すと、もとのテキストが変更されて正しく表示されません。

オーガナイザ

- 日本語バージョンでは、高速検索機能 (Ctrl + f) の使用時はASCII入力のみがサポートされます。

SDLエディタ

- ダイアグラム名は最大255文字です。
- 赤と青を使用したシンボルがある場合、Diff操作ではこの2色が2つのダイアグラムの差異の強調表示に使用されるため、Diff操作の結果表示される図式ビューの解釈が困難になることがあります。この問題を回避するには、ダイアグラムを比較するときに、赤と青をシンボルに使用しないようにします。
- **Internal input**と**Internal output**を表す特殊な図式シンボルは使用できません。Z.100ではこれらの使用を認めていません。通常の入力および出力のシンボルで同じ動作を表すことができます。
- 図式マクロ (マクロダイアグラム) には、分岐のないシングルフローのフローシンボルのみを含めることができます。図式マクロ呼び出しは、プロセス (システム、ブロックなど) より上位のスコープでは使用できません。
- オプションシンボル(破線ポリゴン)は使用できません。
- 入れ子のダイアグラムはサポートされません。つまり、ダイアグラムには、下位のダイアグラムに対する参照シンボルを含める必要があります。たとえば、1つのブロックを持つシステムは2つのダイアグラムとして描く必要があり、そのシステムダイアグラムにブロックダイアグラムに対するブロック参照シンボルを含めなければなりません。
- 環境設定 *Editor*AlwaysNewWindow* の使用時に search コマンドなどを使用すると、新しいウィンドウが表示されます。ダイアログが開かれている間、新しいウィンドウのメニューバーは淡色表示されますが、コマンドによっては選択可能なものもあります。ダイアログが表示されている間は、これらのコマンドを選択しないでください。

印刷

- FrameMakerやInterleafとの統合機能では、色はサポートされません。

- SDT*[PrintFontFamily](#)設定を使用して、テキストファイルの印刷に使用するフォントを変更することのみが可能です。
- HTMLに出力する場合、ダイアグラムのページ名に「¥」は使用できません。

Emacsとの統合機能(UNIXのみ)

- 正しく機能するためには、SDLはGNU Emacs 19.31以上であると想定しています。
- リンクを操作するコマンドでは、[元に戻す]はサポートされません。
- EmacsをSDL Suite and TTCN Suiteから起動して、SDL Suite and TTCN Suiteのみをシャットダウンした場合、そのEmacsのセッションを別のSDL Suite and TTCN Suiteセッションに再接続することはできません。

Microsoft Wordとの統合機能(Windowsのみ)

- MS Word 2003およびWord2000がサポートされます。「動作保証マトリックス」を参照してください。
- MS Wordドキュメントを変更しても、オーガナイザではアイコンにmodifiedのマークが付きません。
- MS Wordの新規ドキュメントには、オーガナイザで与えられていた名前に関係なく、必ず「Document<x>」(<x>は整数)という名前が付きまます。
- リンクを操作するコマンドでは、[元に戻す]はサポートされません。
- 場合によって、MS Word文書の最後は、リンク マネージャで手作業による同期をとる必要があります。*Update*メニュー選択を使用します。
- オーガナイザでは、MS Wordの複数のインスタンスを扱うことはできません。
- NTFSのファイルやディレクトリに対する読み取り/書き込みセキュリティアクセス権の認識に問題があるために、MS Wordとの通信で奇妙な動作が発生することがあります。この場合、ほぼ同じ内容の2つのエラー ボックスが、1つはMS Word側に、もう1つはオーガナイザ側に表示されます。オーガナイザでは、そのようなファイルのアクセス権が、誤って「rw」とマークされています。
- 実装の部分はMS Wordのマクロで処理されるので、ほかのテンプレートやマクロと組み合わせると競合が生じる場合があります。この問題については個々に検討する必要があり、異なるマクロを結合しなければならなくなる場

合もあります。この実装では、たとえば、Microsoft社のMacro Virus Protection Toolは機能しません。

- MS WordをSDL Suite and TTCN Suiteから起動して、SDL Suite and TTCN Suiteのみをシャットダウンした場合、そのMS Wordのセッションを別のSDL Suite and TTCN Suiteセッションに再接続することはできません。

SDLアナライザ

ジェネリック関数

SDL Suite 4.2以降、SDL演算子のコードを生成するデフォルトの方法が変更になっています。この方法により、ジェネレータで生成される代入、等価テスト、解放および定義済み演算子のための関数はジェネリックな関数に置き換えられます。

これにより、パラメータの渡し方も変わります。パラメータは値ではなくアドレスで渡され、データ型について記述したデータ構造が必要になります。

この方法は、インラインCコードを使用するSDLシステムとは下位互換性がありません。関数の呼び出しを手動で変更する必要があります。

実装の制限事項

アナライザの現行バージョンはSDL-92アナライザです。ここでは、その制限事項と実装の制限について説明します。対策があれば、それについても説明します。

- マクロおよびマクロ呼び出しはPRに変換され、PRフォームで展開されます。これによって、マクロの使い勝手が制限されます。たとえば、入口と出口の数は0または1でなければなりません。障害が発生しやすいケースの1つは、分岐から呼び出される状態付きマクロです。障害が発生するのは、GR-PRコンバータがそのマクロの内容を認識せずに、呼び出しをタスクシンボルのように扱うためです。この理由のためにサポートされないものとしては、これ以外に変数宣言と型定義があります。
- アナライザは、外部プロパティの配置を以下の例のように想定しています。つまり、外部の形式名がそれぞれ代替キーワードと同じ行にあり、外部データの記述がそれら自身の行にあるとしています。

例1

```
alternative a,b;  
  abc abc t5;  
  abc abc t5-2;  
endalternative;
```

b;

PR-GRコンバータ

- PRシステムは、SDLのダイアグラムの処理方法に合うように、参照されるすべての定義を持つように変更されます。名前および型がシステム内で一意でない場合、参照される定義には限定子が必要です。そのような限定子は、自動的に提供されません。オーガナイザで *Update Heading* コマンドを使用してください。
- GR形式に変換される間に、PRファイル内のコメント `/* comment text */` がスキップされる場合があります。

構文分析における制限事項

- チェックは、コメントのみのタスクシンボルに対しては行われません。このため、内部にコメント「`/* ... */`」のみを含むタスクシンボルがある場合は、コンパイルエラーが発生することがあります。
- アナライザは、名前の中のスペースを処理できません（スペースは下線の簡略記法です）。空白は明示的な下線で置き換えてください。

意味分析における制限事項

- 仮定型の意味が不明確です。
- 暗黙の信号ルートでのゲート制限がチェックされません。
- 暗黙のゲート接続がチェックされません。
- グローバルプロシージャの信号が正しくチェックされません。
- テストは出力 `via` のみに限られます。
- インスタンス化した型がゲート制約に信号を持つかどうかは、チェックされません。
- コンテキストパラメータは実装されません。
- 条件付き等式は実装されません。
- 入れ子になった仮想定義では、一部のチェックが行われない場合があります。
- USEステートメントのパッケージインターフェイスおよび定義選択リストは実装されません。

- インポート プロシージャ仕様に対するリモート プロシージャ コールはチェックされません。
- エクスポート プロシージャに対するインポート プロシージャ仕様はチェックされません。
- インポートされる変数仕様に対するインポート式はチェックされません。
- エクスポートのプロセスに対するインポートの変数はチェックされません。
- サブタイプおよびスーパータイプのサブストラクチャ名が同じかどうかはチェックされません。これは、チェックを行うと、「ブロック内ブロック」を継承に対して使用できなくなるためです。
- 演算子のシグネチャに対する演算子の定義はチェックされません。
- ユーザー定義のデータ型の名前クラス リテラルは、意味アナライザで扱うことができません。名前クラス リテラルは、正規表現で定義されるリテラル名の（無限の可能性がある）セットを表す簡略記法です。リテラルを明示的に列挙するか、定義済みのデータ型からの継承を使用してください。
- 分岐における質問と回答は常に形式的なものと考えられます。したがって、非形式的テキストは文字列と見なされます。
- データ型定義からの継承にその型定義を参照する限定子が含まれる場合、この継承は使用できません。これは、継承の展開中に限定子がコピーされて、変更されないためです。
- システムの要素を分析するときに、省略された要素に依存するチェックは実行されません。
- サブストラクチャにプロセスとブロックの両方が含まれる場合は、どちらを実装するか選択できなければなりません。ただし、プロセス定義が分析されないため、この選択を行うことができません。
- 信号の詳細化はサポートしていません。

実行されない意味チェック

以下のSDL意味規則の分析は実行されません。

- 完結部分集合は完結詳細化集合でなければなりません。
- 分岐の回答は相互に排他的でなければなりません。このチェックは、一部のデータ型に対して実行されない場合があります。

- エクスポートされる変数は、インポートされる変数のそれぞれに対して存在しなければなりません。
- アクシオムにおける型チェック、および同等クラスの評価とチェックは実行されません。
- ジェネレータ名およびジェネレータの仮パラメータを、限定子の中で使用したり、限定したり、感嘆符の前に置いたり、デフォルト割り当ての中で使用することはできません。

CAdvanced SDL to Cコンパイラ

- ブロックタイプまたはプロセスタイプがあり、分離がユーザーによって指定されている場合、そのブロック/プロセスタイプを継承する他のすべてのブロック/プロセスタイプにも分離がユーザーによって指定されている必要があります。これ以外の場合は、hファイルがインクルードされる順序が混乱することがあります。
- ユーザーが分離を指定し、XFREESIGNALFUNCSが定義されている場合は、コンパイルエラーが発生します。ユーザー指定の境界が使用されているため、信号s_c2b_1がファイルBlockB.hに定義されます。system.cファイル内のcm_Free_Signalの定義では信号s_c2b_1を使用しますが、この信号はインクルードされていないファイルに定義されているため、不明な信号になります。
- チャンネルサブストラクチャは使用できません。
- 信号の詳細化はサポートされません。
- コンテキストパラメータはサポートされません。
- タイマactive式は、パラメータ付きタイマには使用できません。
- 生成したコードでは、delay付きチャンネルとNODELAY付きチャンネルの違いはありません。
- IntegerとRealは、それぞれCのint型（32ビット整数）とdouble型を使用して実装されるので、範囲および精度に制限があります。CharstringはC言語のchar *を使用して実装され、char *はNUL (=0) で終わるため、Charstringに文字NULを含めることはできません。
- アクシオムおよびリテラルのマッピングを抽象データ型定義の要素にすることは可能ですが、その情報はSDL to Cコンパイラでは使用されません。

- 抽象データ型では、以下の概念を扱うことができません。
 - － 名前クラスリテラル
 - － **Charstring**リテラルを使用したリテラルの命名
- ソート定義は、直接的であれ間接的であれ、自身を参照することはできません。
- 構造体のコンポーネントは、生成したコードにおいて、**SDL**における名前と同じ名前を持ちます。ただし、アルファベット、数字、およびアンダスコア以外の文字は削除されます。名前の残りの部分は、以下のような、**C**コンパイラが受け入れる有効な**C**言語の識別子でなければなりません。
 - － アルファベットで始まること。
 - － **C**言語の予約語と一致しないこと。
 - － その構造体を表す**C**言語の構造体内で固有であること。
- 特定の分岐式の値に対する分岐からの複数のパスがチェックされません。シミュレーションプログラムは、存在するパスの中から1つを選択するだけです。
- 整数値および実数値のオーバフローが**C**言語のレベルでチェックされます（実際の**C**言語のシステムがこれらのチェックを実行する場合）。
- プロシージャをエクスポートするプロセスは、状態を含むグローバルプロシージャを、直接的であれ間接的であれ呼び出すことができません。
- **C**がブロックインスタンスセットになる場合、**Output VIA ALL C**は正しく扱われません。ブロックインスタンスごとに1つの信号が送られる必要がありますが、送られる信号はたった1つです。
- **.ifc**ファイルに、その**.ifc**ファイルが実行可能な種類の**SDL**単位に対する**#define**定義が含まれます。すべての**SDL to C**コンパイラはこの原則に従います。**C**コンパイラによっては、これが、コンパイル時のエラーや警告、あるいは実行時のエラーにつながる可能性があります。たとえば、以下のよう
に問題が発生します。

```
newtype x
  literals green, blue;
endnewtype x;

newtype y
  literals blue, green;
endnewtype y;
```

リテラルの**green**と**blue**は、次のように生成されます。

```

/* For newtype x */
#define green 0
#define blue 1

/* For newtype y */
#define blue 0
#define green 1

```

すると、通常、Cコンパイラは、マクロのblueとgreenが別の値で再定義されているというメッセージを表示します。どのような場合でもコンパイルを続行して実行形式を生成するCコンパイラもあります。その結果、実行時エラーが発生しますが、これは、greenおよびblueが共に最初のnewtype xで使用されるときに、2番目のnewtype yからのリテラル値がCコンパイラによって使用されるためです。この問題を回避するには、以下のように、各リテラルに接頭辞を手動で追加する必要があります。

```

newtype x
  literals x_green, x_blue;
endnewtype x;

newtype y
  literals y_blue, y_green;
endnewtype y;

```

このgreenとblueの例は、継承した構文を使用すれば回避できる問題なので、好例とは言えません。newtype xからのリテラルは、newtype yに継承することができます。複数のエンティティクラスがCコンパイラから組み合わされると、さらに問題が発生します（たとえば、greenという信号が#defineとしても生成される場合など）。接頭辞を手動で追加して、.ifcファイルにおける名前の競合を回避することを強くお勧めします。

- **Syntype**にa:bという形式（つまり、1つの制限範囲）の範囲条件が1つでも含まれていると、**any(Sort)**のみが実装されます。ここで、**any(Sort)**の"Sort"の部分はsyntypeが入ります。**Real**や**Time**など、範囲条件を持つ**real**型のsyntypeの場合は、実装されません。
- **any(Real)**は、ランダムに十進値を生成せず、すべての数値を生成するだけです。
- 環境変数のSDT*RelativeSDTREFをオンにしている場合で、別のファイル名で[境界の編集]を使用しているときには、要求しているファイルが存在しないという内容のコンパイルエラーが発生することがあります。この問題は、別の名前を使用しないか、**Relative SDT References**をオフにすることで解決できます。

ターゲティング エキスパート

- ターゲティング エキスパートを使用して、配置図に基づいてコードを生成する場合、オーガナイザの「ユーザー指定の境界」情報は無視されます。
- ソースSDLのパスにスペースがあるかどうかによって、生成されたCファイルの名前は異なります。パスにスペースが含まれている場合、名前は「<systemname>.c」になり、それ以外の場合は「component.c」になります。
- **Cadvanced**ライブラリに対する自動および手動設定/スケーリング機能は、すべての定義済みの統合でデフォルトに設定されているわけではありません。コンパイラ フラグ-DUSER_CONFIGを、設定情報を使用するように設定する必要があります。

ターゲティング エキスパートの [ホスト接続] セクションのすべてのタイプについて、アライメントを 8 に変更して解決します。つまり、「ターゲット データ コーディング: アライメント」ですべてのエントリを 8 に設定します。これらの値は、sdtmt.opt ファイルから読み込まれます。

- SET OFの使用時は、関数GenBag_Take2でメモリ リークが起きます。

メモリ リークを発生させる例は以下のとおりです。

```
y:= Take(test_set, 1)!item
```

CPP2SDL

- 以下のC/C++言語は、ツールでサポートされます。
 - ANSI C/C++
 - Microsoft Visual C/C++
 - Gnu C/C++
- このツールは、入力C/C++プログラムの完全な文法分析を実行しますが、意味論分析が行われるだけです。古いKernighan & Ritchieスタイルの関数宣言はサポートされません。
- C++の例外はサポートされません。
- C++テンプレート宣言はSDLに直接翻訳されません。その代わりに、テンプレートのインスタンス生成を提供する必要があります、これらは翻訳されます。
- 関数ポインタのサポートには制限があります。関数ポインタ型は、SDL (型なしポインタ) に表示されますが、たとえばポイントされた関数を呼び出すことはできません。

- `sizeof` 演算子を使用している式は常に正しく評価されません。
- C++ `cast`式は完全にサポートされません。
- 仮ベース クラスから継承されたメンバーは、SDL翻訳にアクセスできない場合があります。
- 同じテンプレートでの複数のテンプレートのインスタンス生成から継承するクラスは、完全にサポートされません。
- 特定の条件が問題を引き起こしている状態では、「`sizeof`」は正しく処理されません。
- たとえば、匿名型を使用して以下のように記述した場合、問題が発生することがあります。

```
typedef struct {  
    int i;  
    float f;  
}mess;
```

以下のように記述してください。

```
typedef struct mess {  
    int i;  
    float f;  
}mess;
```

SDLシミュレータ

- `add-macro`機能と組み合わせて配列または集約を実パラメータとして渡すと、実行中にパラメータ情報が失われたり、誤って解釈されることがあります。パラメータは入力スクリプトに正しく渡されません。これは、複数の文字を一重引用符で囲んで渡した場合も同様です。マクロを使用する代わりに、明示的なパラメータを渡して入力スクリプトをパラメータ化すると、この問題を回避できます。
- シミュレータが限定子を処理する方法には制限があります。これが原因で、アナライザはブロックの2つのインスタンスが同じ名前であることを検出できないことがあります。このため、シミュレータ実行形式を問題なく構築することはできますが、インスタンスの名前が不明確なために変数のチェックなどが不可能な場合があります。
- オーガナイザで、SDLシミュレータをすでに生成しているときに、別のオプションや異なる部分を選択して新しいシミュレータを生成する場合には、[フル実装]を実行する必要があります。これは、オプションの変更や選択の変更

が生成プロセスによって処理されないためです。フル実装を実行しないと、生成プロセスにおいてコンパイルエラーまたはリンクエラーが出力されることがあります。

- 通常の配列には、範囲チェックが実装されません。通常の配列は、リンクの一覧として実装されます。これは、コードジェネレータでは、一般的な配列として実装できるかどうかを判定できないためです。この問題は、変数に変換されるシノニムがsyntypeの範囲条件に使われ、そのsyntypeが配列やpowersetのインデックスソートとして使われている場合に起こります。この理由は、配列の長さの指定にC言語の変数を利用できないためです。

以下に例を示します。

```
SYNONYM Lo NATURAL = 1;  
SYNONYM Hi NATURAL = 10;  
SYNTYPE BufferType = NATURAL  
CONSTANTS Lo , Hi  
ENDSYNTYPE;  
NEWTYP MessageBufferType  
    Array( BufferType, CharString)  
ENDNEWTYP;
```

エクスプローラ/オートリンク

- Verify MSC コマンドを使用する場合、MSC ファイルのファイル名またはパスにスペースを使用できません。
- ポインタも含む構造体のビットフィールドは扱いません。
- C共用体にポインタを含めることはできません。
- エンコード/デコード機能をエクスプローラ カーネルと一緒に使用することはできません。
- MSCの検証には、ptr_charやCharstarデータ型のパラメータを送信する信号へのチェックが含まれており、「No reference to dynamically allocated memory.」というエラーが発生します。

MSCダイアグラムでの読み取りのとき、Charstarデータ型にパラメータが存在し、これを読み取ろうとすると正しく処理できない動的なメモリができてしまうため、参照エラーが発生します。

例

- UNIXでは、Phoneアプリケーションがコンパイルされた後にApplicationDebugカーネルとリンクされる場合、phone の例を図式シミュレータUIと共に使用することはできません。

リアルタイム オペレーティング システムのサポート

ライト インテグレーション

SDL to Cコンパイラのものと同じ制限事項が適用されます。 [23ページの「CAAdvanced SDL to Cコンパイラ」](#)を参照してください。

スレッド インテグレーション

SDL to Cコンパイラのものと同じ制限事項が適用されます。 [23ページの「CAAdvanced SDL to Cコンパイラ」](#)を参照してください。

- スレッド インテグレーションは、配置エディタとターゲティングエキスパートからのみ生成できます。つまり、オーガナイザの実装機能は使用できません。
- VxWorks用のスレッド インテグレーションは、POSIXライブラリ関数を使用します。このことは、VxWorksカーネルをコンパイルするときはPOSIXライブラリを使用しなければならないことを意味しています。
- スレッド インテグレーションでは、状態遷移がSDLシステムで発生した場合にxInEnv関数が発行されます。これは、xInEnvを呼び出す間隔は一定ではなく、実行中のSDLシステムのプロパティに依存していることを意味します。SDLシステムに入力する外部コードを書く場合は、システムに信号を挿入するのにxGetSignalとSDL_Outputを使用して、コードが保有するスレッド内でコードが実行されるようにすることをお勧めします。
- オンラインMSCトレースは、VxWorksソフトカーネルの元では動作しません。

TCP/IP通信モジュール

- VxWorksでは、TCP/IP通信モジュールはSolaris上で動作しているSoftKernelでテストされました。Vxworks softkernelアプリケーションに接続するためには、ホストコンピュータのループバックIPアドレスを使用する必要があります。ほとんどの場合、このIPアドレスは「127.0.1.0」です。
- OSEでは、TCP/IP通信モジュールはSolaris上で動作しているSoftKernelでテストされました。OSE SoftKernelアプリケーションからリモート コンポーネントに接続するには、リモートホストのIPアドレスを明示的に指定する必要があります。「the_computer.the_company.com」などの標準のホスト名では不十分です。

タイト インテグレーション

- **SDL Suite**のパッケージに含まれるタイト インテグレーションは、**Cadvanced**のみに適用されます。**Cmicro**には適用されません。
- **Solaris (Posix 4)**、**Win32**、**VxWorks**、**Nucleus**、および**VxWorks OSE** デルタ統合のみがダウンロード可能です。
- 優先入力は処理されません。
- コンティニューアス信号は処理されません。
- サービスは処理されません。
- パラメータを持たないプロシージャを返す値の**RPC**呼び出しは機能しません。
- **SDL**の**ANY**構文は正しく実装されません。
- タイマセット構文**SET(5, T1)**はサポートされません。代わりに**SET(NOW+5, T1)**を使用してください。
- `$stelelogic/sdt/include/ADT`には、以下の**ADT**ライブラリ ファイルがあります。

- `file.pr` :

RTOS環境でサポートされる場合に機能します。一部のコンパイラでは、コンパイルできるようにするために、`xGetValue`関数を以下のコードで囲む必要があります。

```
#ifdef XMONITOR
...
#endif
```

- `list1.pr`および`list2.pr` :

これらは、メモリの消失につながる動的ポインタと共に実装されるため、使用しないでください。

- `byte.pr` :

正しく動作します。

- `unsigned.pr`、`unsigned_long.pr`、`longint.pr` :

正しく動作します。

- `pidlist.pr` :

正しく動作しますが、出荷されているRTOSで提供されているバージョンに限られます。

す。`$stelelogic/sdt/sdtdir/RTOS/INCLUDE/pidlist.pr`

– `idnode.pr` :

`FirstPid`関数および`NoOfProcesses`関数は、特定の条件下で動作します。これ以外の関数は動作しません。

– `pointer.pr` :

メモリを消失する危険性が高いので使用しないでください。

- **Kernighan & Ritchie C**のサポートは終了しました。
- 統合コードは、**SDL**システムで宣言されたプロセスインスタンスの数の上限をチェックしません。
- 同じデータ型が、**TIME**および**DURATION**を表すために使用されますが、ほとんどのオペレーティングシステムでは、これは符号なしの整数が使用されます。したがって、**TIME**式や**DURATION**式は負の値として評価されません。また、これは中間式の結果にも適用されます。
- **SDL**エラータームは、**Cadvanced**タイトインテグレーションではサポートされません。たとえば、「`return error;`」は、**Cadvanced**タイトインテグレーションでは使用することができません。
- **REVEAL**および**VIEW**はサポートされません。この概念を採用しているアプリケーションはクラッシュする場合があったり、同時性問題のために定義されていない動作が発生する場合があります。
- **EXPORT**および**IMPORT**はサポートされません。この概念を採用しているアプリケーションはクラッシュする場合があったり、同時性問題のために定義されていない動作が発生する場合があります。
- さらに、**SDL-C**コンパイラの場合と同じ制限事項も適用されます。[23ページの「CAdvanced SDL to Cコンパイラ」](#)を参照してください。

SDL Cコンパイラ ドライバ(SCCD)

- 制限事項：SCCDをDOSやWindowsで使用する場合、二重引用符で囲まれたインクルードパスは処理されません。

Cmicro SDL to Cコンパイラおよびライブラリ

- 異なるブロックにあるプロセスは同じ名前にしないでください。これは、**Cmicro**にシンボル ツリーが存在しないためです。たとえば、(SDL環境の) 同じCモジュール内で、同じ名前を持つ2つのプロセスをアドレス指定することはできません。これを回避するには、システム内の異なるプロセスには常に別の名前を付ける必要があります。各プロセス名に接頭辞を追加すれば、簡単に別の名前を付けることができます。
- 定義済みのソート `carray` :

SDLプロシージャから定義済みのソート `carray` の変数を返すことはできません。これは、ANSI C言語ではこのソート `carray` を扱うことができないためです。ただし、回避策として、この `carray` を `newtype` 構造体の値に含めることはできます。
- パラメータ付きのタイマとプリエンブティブなCmicroカーネルを同時に動作させることはできません。
- プロシージャが、アドレスとして渡されるソートの値を返す場合、`prd(invar, retvar);` の呼び出しのような代入はサポートされません。この代わりに、`retvar := call prd(invar);` という記述を使用します。
- シントypesの名前には以下を使用しないでください。

```
int
unsigned_int
long_int
unsigned_long_int
short_int
unsigned_short_int
char
signed_char
unsigned_char
float
double
ptr_void
bool
wchar_t
```

Cextreme SDL to C コンパイラ

以下の項目はCextreme SDL to C コンパイラではサポートされません。

- サービス
- 状態をもつプロシージャ

- 外部プロシージャのDCLまたはFPARへのアクセス
- VIEW/ REVEAL
- コンティニューアス信号
- 優先入力と優先出力
- #PRIO
- チャネルサブストラクチャ
- 2つ以上のパラメータをもつタイマー
- ソートInteger以外のパラメータをもつタイマー
- プロシージャまたは演算子内のリモートプロシージャコール
- 仮想プロシージャ呼び出しの場合の限定子の使用

SDLターゲット テスタ

- (中断コマンドを送ることによって) ターゲットシステムが中断した場合、ターゲットに送られる信号はすべて破棄されます。
- Output-PARまたはOutput-NPARコマンドを使用してターゲットシステムに信号を送信する場合は、そのシステムで定義されるすべての信号を含む一覧が提供されます。SDLターゲット テスタの出力コマンドのパラメータを指定するときは、必ず正しいプロセス インスタンスを対象にしなければなりません。
- Output-PARコマンドを使用して文字列またはポインタをターゲットに送ることはできません。
- 複数のコンパイラ (たとえば、Tasking 80166) は、メッセージ長について正しくない情報を提供します。

このエラーはSDLターゲット テスタによって検出され、一度だけ表示されます。SDLイベントのトレースが正しいのは、後続の通信にCRCチェックが含まれない場合だけです。

- SDLターゲット テスタは、ターゲティングエキスパートの [ツール] メニューからのみ起動してください。SDLターゲット テスタがスタンドアロンで起動する際に問題が発生する可能性があるからです。

- **SDL**ターゲット テスタのページファイルおよび変換ファイルは動作しません。
- **Unix : Cmicro**ポストマスタは、一時ファイルを `/tmp` の下に保存します。これらのファイルはとどきとき手作業で削除する必要があります (`rm/tmp/.sdt.cmicro.*`)。
- タイマ入力に対してブレークポイントを指定することはできません。
- **Cmicro**ポストマスタのデバッグ モードを使用すると、長すぎるメッセージが発生することがあります。その場合、ポストマスタは、コア ダンプによりクラッシュします。
- **SDL**ターゲット テスタの行番号機能をオンにして再度オフにすると、テキスト領域の1列目が分割される場合があります。これを回避するには、常に行番号機能を最初にオンにして、その後はオフにしないでください。
- コンテキスト対応ヘルプにおいて、現状では使用できないコマンドが使用可能であると表示される場合があります (ターゲット システムが接続されて動作していない場合でも、ブレークポイントを設定することはできます)。
- **UNIX : sdtmt.btn**ファイルに含まれる「:MENU」とマークされたエントリが多すぎると、**SDL**ターゲット テスタは起動時にウィンドウのサイズを調整できません。ウィンドウのサイズを手動で変更すれば、正しいサイズになります。
- パラメータのトレースは、**Predefined Sorts, Syntypes, Enums, Newtype Stucts**、および**Newtype Array**に限定されます。
SDLターゲット テスタが不明なデータ型を検出した場合は、残りのパラメータは**hex-buffer**として表示されます。
制限事項：
Charstringの内容を表示することはできません。
`sdtmt.opt`ファイルで**LENGTH_**、**ALIGN_**、および**ENDIAN_**に誤った値を使用すると、テスタはすべてのパラメータをデコードしますが、表示される値は正しくありません。
- 既定では、正規表現を使用して、ターゲット テスタのテキストのトレースを検索することはできません。トレースの最後に到達するとテスタはクラッシュします。このことは、そのトレースには正規表現がそれ以上存在しないことを意味します。
ターゲット テスタを開始する前に、環境変数**TT_REGEX**を1に設定することでこの機能が使用可能になります。

Cmircroを使用したアプリケーション生成

- ターゲットエキスパートで、発生しうるすべてのエラーがチェックされるわけではありません。ほとんどの場合、コンパイル時エラーは、設定が正しくないために発生します。正しく設定するには、オンラインヘルプを参照してください。

- `ml_typ.h`で`#include "systemname.ifc"`を実行しようとしたために、あるいは、これ以外の理由で、名前の競合がさらに発生することがあります。たとえばプリプロセス中に、次のエラーが発生します。

```
sctpred.c:In function `yAss_SDL_Object_Identifier':
ml_pred.c:2109:parse error before `0'
ml_pred.c:2109:parse error before `0'
```

上記のエラー（および、その他の予測できない動作）は、“`SIn`”という信号があったために発生しました。これは、`#define SIn 0`として生成されます。一方、`sctpred.c`ファイルには、`SIn`という変数があります（マクロの`yAss_SDL_Object_Identifier`の中）。この場合、Cプリプロセッサが変数名を`SIn`の`#define`値で置き換えるため、上記の解析エラーになります。

この問題を回避するには、コード生成されたヘッダーまたは`.ifc`ファイルが**Cmircro**ライブラリ自体や生成したCコードのどの部分にも含まれないようにします。`.ifc`ファイルは、SDL環境ファイル（`env.c`など）のみに含める必要があります。

- Cコードを生成すると、未使用の各変数に対して次の警告が通知されます。

```
"Unused Variable :<variablename>"
```

Cコードの生成で発生したこれらの警告は簡単には削除できません。なぜなら、コード生成中に警告を処理するのが困難であるためです。これらの警告が実際に問題になることはありません（ただし、Cコンパイラから出力される大規模なファイルは読み取れない場合があります）。以下の変数が参照されることがあります。

```
warning:unused variable 'yOutputSignal'
warning:unused variable 'yVarP'
warning:unused variable 'ySVarP'
```

パラメータのない信号を使用すると、Cコンパイラが上記と同じ警告を発生します。

次の警告は、`mk_outp.c`モジュールで容認されます。

```
mk_outp.c:warning:unused variable 'P_Parameter'
```

次のような同様の警告も、mk_stim.cで容認されます。

```
mk_stim.c:warning: 'SystemTime' defined but not used
```

- 警告例：「Control reaches end of non-void function」などの警告

基本的に、SDL to CコンパイラがC言語のswitch()構文を生成し、PAD関数（プロセスアクティビティ記述関数）からの戻りはすべて、そのスイッチのすべてのケース分岐で発生するため、スイッチの後のreturnステートメントがPAD関数に含まれません。このため、一部のCコンパイラは「Control reaches end of nonvoid function」と表示します。この警告は容認できます。ml_typ.hファイルやuser_cc.hファイルにある警告は、END_PADマクロを次のように変更することで削除できます。

```
#define END_PAD(PrsNameWithPrefix) return  
(SDL_DASH_NEXTSTATE);
```

- 警告：xSTATE_INDEX_zyyyyy_xxxxxxx[0]のイニシャライザを囲む中
かっこの欠落

これは、これまでのところ、主要なCコンパイラでは正しく処理されているので、容認できるエラーです。SDL to Cコンパイラがこのような警告を発するのは、ANSI Cだけでなく、Kernighan & Ritchie Cにも準拠している必要があるためです。

- Cコンパイラによっては、非常に長いマクロ定義のプリプロセス時に失敗する場合があります。たとえば、8051ファミリのCPU向けに販売されているすべてのCコンパイラなどです。そのような長いマクロ呼び出しは、定義済みのジェネレータ（String、Powerset、Bag）で発生します。

Cコンパイラ（または、特にCプリプロセッサ）が、マクロ定義が長すぎるという意味のメッセージを表示したり、誤りのある出力を生成する場合は、これらのジェネレータを使用できません。マクロのX_LONG_MACROSは、ターゲットエキスパートを使用して設定してください。

- 組み込みシステム向けのCコンパイラによっては、再帰的なマクロ呼び出しをコンパイルできない場合もありますたとえば、再帰的なマクロ呼び出しは、SDLステートメントでは次のように表示されます。

```
a:=b and c and d;
```

そのような問題が発生する場合は、たとえば以下のように設定して、問題の起こるステートメントを使用しないよう強くお勧めします。

```
tmp:=b and c, a := tmp and d;
```

- ターゲットアプリケーションに関しては、メモリの割り当て方法（テンプレートディレクトリにあるCモジュールmk_cpu.cを参照）、およびメモリが不足したときの動作を常に指定する必要があります。テンプレートCファイルはユーザーの意図を認識しないので、NO_MORE_MEMORYと呼ばれるCマクロをユーザーが定義する必要があります。このような致命的なエラーが発生した場合、プログラムの停止や、リセット、またはエラーメッセージの表示などを行うことが適切です。ただし、表示機能にもメモリが割り当てられている場合は、エラーメッセージの表示も失敗する可能性があります。
- XMK_USE_KERNEL_INITが定義されていると、プロシージャ内で変数のデフォルト値が設定されません。これは、簡単には解決できない問題です。

対策としては、以下の2つがあります。

- XMK_USE_KERNEL_INITを定義しません。
- プロシージャ内では変数にデフォルト値を使用しないでください。たとえば、以下の記述はプロシージャ内では許可されていません（プロセス内では正常に動作します）。

```
DCL a integer;
DCL b integer;
task b:=a+b; /* 結果が確定していません */
```

以下のように記述してください。

```
DCL a integer = 0;
DCL b integer = 0;
task b:=a+b; /* 結果が確定しています ! */
```

CmicroのSDLにあるADTの使用

SDLに付属するパッケージおよびADTは、主にCbasic/Cadvancedアプリケーションで使用されます。ただし、これらのADTには、Cmicroアプリケーションに役立つものもあります。それ以外のADTには、Cbasic/CadvancedのCコードへの参照が含まれるので、Cmicroと共に使用することはできません。禁止事項と推奨事項の一覧については、『[User's Manual](#)』の第65章「[The Cmicro SDL to C Compiler](#)」の3378ページ、「[Abstract Data Types](#)」を参照してください。

Cadvanced / Cmicroのコードの結合

異なるSDL to CコンパイラのCコードを組み合わせることはできません。これは、SDL to Cコンパイラがそれぞれ独自の実行時モデルと実行時データ構造を使用するためです。これらのCコードを組み合わせると、いずれコンパイ

ル時エラーが発生します。この制限は、`sdth2sdl`を含め、あらゆるCコードの組み合わせに当てはまります。

TTCN Suite ツールの制限事項

Solarisでの色の問題

- コンピュータのディスプレイで同時に表示できる色の数に制限がある場合は、**TTCN Suite**で色が不足することがあります。この場合、**TTCN Suite**は無作為に色を表示します。この現象は**TTCN Suite**の使い勝手には影響ありませんが、多少いらいらするかも知れません。

Solarisでのテーブルエディタ

- テーブルエディタでは、約1000行までしか表示できません。この制限はテーブルエディタにのみ存在し、MPとの変換や、印刷、分析などには影響はありません。

アナライザ

- **BMPString**、**UniversalString**の2つのタイプには対応していません。
- アナライザでは、エンコード/デコードの変換はサポートされません。
- **ASN.1** 列挙型識別子の先頭には、大文字と小文字の両方を使用できます。これは **TTCN Suite** ツールに対してのみ適用されるものであり、**ASN.1** ユーティリティ ツールには適用されません。
- 4980: アナライザは、`send` または `receive` ステートメントに続いて空の **behavior** ラインがあると不完全な分析しか行いません。
- **タイプルックアップ**：
タイプルックアップの実装の制約により、タイプが参照されるコンテキストが、ルックアップする値をよりも先に参照される必要があります。
以下に例を示します：

```
[{a 2} = variable_with_sequence_as_type]
```


この例では、シーケンス値のタイプのルックアップに失敗します。
回避策：先に値を参照するように書き直します。

```
[variable_with_sequence_type = {a 2}]
```

外部ASN.1参照

参照されるASN.1型の定義には、以下の制限があります。

- `Module.Type`形式の参照は、定義内では使用できません。
- 同じASN.1モジュールまたは別のASN.1モジュールでのデータ型または値の参照は、別々に定義されない限り、定義内では使用できません。
- TTCNから参照されるデータ型と値の制限事項には、[13ページの「ASN.1ユーティリティ」](#)で説明している制限も含まれます。
- ASN.1 PDU型定義を参照用を使用する場合、PDU名をテストスイートに対してローカルのほかの名前にマップしていると、BERまたはPER ASN.1エンコード/デコードを使用できません。(PDU名は実行システムに認識されません。)

TTCN ASN.1 BERエンコード/デコード

- 以下の複合ASN.1データ型はサポートされていません。
 - EXTERNAL
 - EMBEDDED PDV
 - CHARACTER STRING
 - ObjectDescriptor
 - UTCTime
 - GeneralizedTime
 - TeletexString
 - VideotexString
 - GraphicString
 - GeneralString
 - T61String
 - ISO646String
 - BMPString
 - UniversalString
 - UTF8String (from X.680/1997)

これらのデータ型の値をエンコードおよびデコードすることはできません。

TTCN to Cコンパイラ

TTCN to Cコンパイラでサポートされる言語は、アナライザで処理される言語のサブセットです。つまり、場合によっては、正しく分析された構文がコード生成段階ではサポートされないことがあります。ここでは、現時点で確認されているTTCN to Cコンパイラの制限事項について説明します。

- INTEGERデータ型の範囲と精度には、制限があります。これは、Cの符号なしint（通常は32ビット）データタイプと符号属性を使用して実装されているためです。
- ASN.1値範囲サブタイプのMIN/MAX構文および包含的/排他的（つまり、<）構文はサポートされません。
- INFINITYは、いかなる用途でも、サポートされません。
- 選択型とその値はサポートされません。
- REALデータ型の値はサポートされません。
- エンコード/デコードの変換はサポートされません。
- TTCNの2つの識別子の最後の文字、D、TestCase（テストスイート変数およびTestCaseD）が異なる場合、TTCN Suite は名前の競合するコードを生成することがあります。以下に例を示します。

Name of Test Suite Variable: TestCase

Name of Test Case: TestCaseD

MSC ログ出力

ETSのMSC生成の機能は、スコープ内に限定され、さらに特定の用途に限って有効です。ここで説明する制限事項は、`mscgen.h`、`mscgen.c`、`static.c`、`gci.c`のファイル内のコードを手動で編集することにより回避できます。IBM Rationalでは、この手動での編集はサポートしていませんが、対策案については前向きに検討いたします。

- 分散配置された同時テスト要素のログ出力はサポートされていません。

これは、分散配置されたテスト要素間の通信を追加する必要があるためです。共通のログを取得するこの機能は、TTCNまたはGCIインターフェイスを使用している場合、いかなる種類のログに対してもサポートされません。したがって、分散配置されたETSにMSCログを適用することは事実上不可能です。なお、複合モードは実行可能ですが、ログは一度に1つの要素に対してのみ出力されます。

TTCNの内部同時モードはMSCジェネレータでは完全にサポートされます。

- テストコンポーネントの再起動

1つの要素が複数回生成される同時テストスイートはMSC/PRとして出力され、MSCエディタではサポートされません。

MSCログを生成すると、同じ名前の複数のインスタンスが出力されます。ただし、同時に出力されることはないと考えられます。この問題を回避するには、テキストを処理するスクリプトを記述して、複数回生成されるインスタンス名を変更する方法以外考えられません。なお、この制限は非複合モードにのみ適用されます。

- 値表記の制限

値をMSC/PR形式にエンコードするときには、いくつかの制限事項があります。値のMSC/PRへのエンコードを無効にする方法の詳細については、[『User's Manual』の第32章「The TTCN to C Compiler \(in Windows\)」の1316ページ、「Compiling an ETS with MSC Generation」](#)を参照してください。

- MSCログのイベントを使って値をログ出力する処理は、一部のデータ型に対してサポートされません。サポートされていないデータ型が使用されている場合には、警告メッセージがイベントログに追加出力されます。詳細については、`mscgen.c:MscEncodeValue`関数を参照してください。
- 使用頻度の低い一部のエンコード機能は、SDLシミュレータで生成したエンコード結果と異なる場合があります。この相違点は、`mscgen.c:MscEncodeValue`関数を編集することにより修正できます。なお、この編集はIBM Rationalではサポートしていません。
- SDLでサポートされていない、いくつかのデータ型には、処理できるものがあります。HEXSTRINGと導き出されたTTCNデータ型は、SDLでは認識されません。
- MSC値のエンコードに対して、ツールには固定サイズのバッファが割り当てられます。このため、値によっては途中で切れてしまうものがあります。文字として表示する値の場合には、約4000文字を超えると切れてしまいます。バッファサイズの追加に必要な定義の説明については、[『User's Manual』の第32章「The TTCN to C Compiler \(in Windows\)」の1316ページ、「Compiling an ETS with MSC Generation」](#)を参照してください。バッファが一杯になった場合は、MSC/PRの生成に対する構文エラーが出力される場合があります。
- MSCキーワードと同じ名前のメッセージタイプを使うと、正しくないMSCが出力されることがあります。MSCジェネレータやTTCNアナライザではこのようなキーワードを検出する機能はありません。MSCエディタで、このようなキーワードの問題を持つMSC/PR形式のファイル

を読み取ろうとした場合には、この問題が検出されます。ASP、PDU、CMの名前に、キーワードなどの名前は使わないでください。問題となる名前には、「reset」、「RESET」、「stop」、「msc」、「TimeOut」などがあります。

- 同時TTCNの設定
 - － 生成するMSCで使用されるインスタンスには、テストスイートで定義されているすべてのPCOが盛り込まれます。結果として、あるPCOを特定の設定のテストケースでは使うように定義していない場合でも、生成するMSCに盛り込まれてしまいます。
- MSCエディタのログ出力機能は、特定のエラーに対する制御が十分でなく、長時間テストを実行するとMSCエディタによって大量のメモリが消費されます。また、MSCエディタは各ログ出力に対して余分なりモート プロシージャコールを2回実行するため、テストの速度が非常に遅くなります。このオプションの使用は、テスト スクリプトの開発やテストの用途に限って使用されることをお勧めします。
- 複合モードで使用している要素は、MscInとMscOut関数を編集して反転することができますが、これらの関数に定義されているfromパラメータとtoパラメータは入れ替わりません。表示の反転は、概念としては自然ですが、SDL Suite and TTCN Suite ファミリのツールには使用できないものがあります。
- コンパイル時のデバッグ用にDEBUG_POST_COMMUNICATIONという定義が追加されています。この定義を設定すると、SDL Suite and TTCN Suiteポストマスターとの間で送受信したstderrメッセージが表示されます。[『User's Manual』の第10章「The PostMaster」の486ページ](#)、[「POSTDEBUG」](#)も参照してください。MSCエディタ ログでイベントが出力されていない場合には、処理できないエラー条件が存在するかどうかを判定するために使われていることがあります。
- MSCエディタには、ダイアグラムを同時に開くことのできる数に上限があります。各テスト ケースには個別にMSCダイアグラムが生成されるため、多数のテスト ケースを実行していると、限界に達してしまうことがあります。この制限数に到達した場合の処理は特に定義されていません。したがって、多数のテスト ケースを実行するときには、MSCエディタのログ出力を使用しないことをお勧めします。

TTCNシミュレータ

SDL to Cコンパイラの制限事項はすべてシミュレータにも適用されます。[39ページの「TTCN ASN.1 BERエンコード/デコード」](#)を参照してください。

- ほかのテストコンポーネントを監視する場合、その状態は、メイン画面ログにのみまとめて表示することができます。
- テストスイートの変数とタイマは、変更も表示もできません。
- チャネルの内容は変更できません。
- テストスイート全体を選択して実行することはできません。手動で選択する必要があります（たとえば、すべてのテストグループか、すべてのテストケースを選択します）。
- SDLシミュレータの実行中にTTCN-SDLのもうひとつのシミュレータの実行を中止すると、その後の通信ができなくなることがあります。一方のTTCN-SDLシミュレータの実行を中止した場合は必ずSDLシミュレータを再起動してください。

SDL and TTCN Integrated Simulator

- SDLシステムにジョイン/マージされたASN.1モジュールが、SDL and TTCN Integrated Simulatorでは動作しないこともあります。同じジョイン/マージ機能がTTCN側でも同様に働くとは限らないからです。そのため、名前が衝突した場合の解決方法が、SDLとTTCNとは異なります。

TTCNエクササイザ

タイマ時間の制限

- タイマに約200000000 ($2^{31}-1$) 時間単位を超える時間をタイマ宣言で定義した場合、特にエラーメッセージが生成されないまま誤動作になる可能性があります。
- タイマ時間が、理論上の最大値を超えた場合、特にエラーメッセージが生成されないまま誤動作になる可能性があります。

値のエンコードとデコードの制限

値のエンコードとデコードはほとんどの値を読み取りおよび書き込みできるように汎用的な機能として設計されています。さらに、エンコードとデコード処理が可能な値には、生成するテストスイートのデータ型システムに定義されていない値も含まれます。このため、期待されていないメッセージや、不適切なメッセージ

も入力可能です。さまざまな期待されていないメッセージを入力することにより、テストスイートやカーネルは簡単に異常終了してしまいます。シミュレーションの実行後の変数値を取得するには、メッセージを指定する作業が若干必要になる場合があります。メッセージの指定には、以下の注意事項や制限がこれらの問題を回避するために役立ちます。

- 値のエンコードおよびデコード処理では、エラーチェックが限られた項目のみ実行されるため、エンコード/デコード中に不明確なメッセージが出力される場合があります。また、不適切な組み合わせで使用した場合にはデータが壊れる場合もあります。

たとえば、`asp1`がASP（基となるデータ型はシーケンス）の場合、以下の値をカーネルに入力すると、エンコード時にランタイムエラーやデータの破壊が起こります。

```
asp1 24
IA5String TRUE
```

一般に、カーネルはあらゆるオブジェクトを入力として受け付けるため、この問題をエラーとして検出する機能はありません。通常、上記の問題は、データ型を変更して異なるデータ型のオブジェクトに割り当てた場合に発生します。

- 単純データ型の場合、実際のデータ型の名前はエンコードされません。この部分的な実装によって、エンコードされる値の可読性は向上しますが、同じメッセージをデコードしたときに不明確になる場合があります。

例：`VideotexString "Hello"`という値は、`"Hello"`のみがエンコードされ、実際のデータ型名はエンコードされません。同じメッセージをデコードした場合、オブジェクトはデフォルトのストリングデータ型である`IA5String`として出力されます。実際のデータ型を指定するために、`VideotexString "Hello"`のように実際のデータ型の接頭辞を使用します。

- `ENUMERATED`値は、名前ではなく、整数値としてエンコードまたはデコードされます。
- `CHOICE`値は、タグではなく、値によって識別できるようにする必要があります。これは、TTCNエクササイザのエンコード/デコード処理がタグの使用に対応していないためです。
- `OBJECT IDENTIFIER`型はサポートされません。

同時TTCN

- **TTCN**エクササイズには、同時に有効にできる同時テスト要素の数に制限があります。制限数は、11以上25未満です。この制限の理由は、スレッドスケジューリングの実装には潜在的なオーバーヘッドがあり、スレッドの数によってオーバーヘッドが増加するためです。6から8個まで同時テスト要素を実行する場合には問題は発生しませんが、カーネルの処理速度が著しく低下します。リアルタイムテストでこの問題を対処するには、決定論的スケジューラや純粹なリアルタイムシステムスケジューラなどの代替手法を導入する必要があります。
- 並列テスト要素を使用している場合にテストケースをキャンセルすると、**PTC**スレッドのハングアップやロックなど予期せぬ動作が発生する場合があります。キャンセルする場合は、必ず事前に同時テストケースを**SNAPSHOT**状態にする必要があります。
- すべての**PTC**を停止する前に**MTC**を終了する処理は定義できません。**TTCN**エクササイズによってテストの条件は送信されますが、テスト結果は無効になる場合があります。すべての**PTC**を停止する前に、**MTC**を停止することは**TTCN**の規定に違反しています。
- **TTCN**エクササイズで**CP**の内容を一覧表示する方法はありません。

TTCNリンク

- エンコード/デコード機能は、**TTCN**リンクカーネルとは一緒に使用できません。
- 追加された**ASN.1**モジュールは、**TTCN**リンクではサポートされません。

自動リンク

- 追加された**ASN.1**モジュールは、自動リンクではサポートされません。

保存形式の互換性

システムファイル

バージョン5.1と比較すると、システムファイルのファイル形式（.sdt）は、**SDL Suite 6.2 and TTCN Suite 6.2**で変更されています。**SDL Suite 6.2 and TTCN Suite 6.2**では、古いシステムファイル（3.0X-5.1）を読みとることができますが、4.4より古いバージョンでは新しいシステムファイルを読みとることができません。

古いシステム ファイルをオーガナイザで保存すると、新しい形式に変換されます。

SDL/GRおよびMSC/GR

バージョン4.1では、**SDL**ダイアグラムの保存形式が変更されました。つまり、4.1で作成されたダイアグラムは、それ以前のバージョンでは開くことができません。SDL Suite 4.1Jは、すべての古いバージョンの形式を読みとることができます。

バージョン4.0では、**SDL**ダイアグラムの保存形式が変更されています。つまり、4.0で作成されたダイアグラムは、それ以前のバージョンでは開くことができません。

SDT 3.1Xにおける**SDL**および**MSC**のダイアグラムの保存形式は、**SDT 3.0X** (および**SDT 2.X**) から変更されました。**SDL Suite and TTCN Suite**の3.1X以降のバージョンの**SDT**では古いファイルを読みとることができますが、以前のバージョンの**SDT**ではバージョン3.1Xから採用されている新しい形式を読みとることはできません。

また、バージョン3.5では**MSC**ダイアグラムの保存形式が変更されました。バージョン3.5および3.6の**MSC**エディタでは、バージョン3.4のエディタと同じ形式を処理できますが、バージョン3.4以前の**MSC**エディタでは新しい**MSC**形式を読み込むことはできません。

SDT 2.X形式および**SDT 3.0X**形式で保存されている**SDL**ダイアグラムや**MSC**ダイアグラムは、そのまま**SDT 3.1X**以降のバージョンのエディタで開くことができます (**SDL**エディタの[開く]などのコマンドを使用します)。ただし、ダイアグラムの本来の構造を保持するには、ダイアグラムをオーガナイザにインポートする必要があります。そのためには、[**SDL**のインポート (*Import SDL*)]コマンドを使用して、システム ファイルに保存できるダイアグラム構造を作成します。

開いている**SDT 2.X**または**3.0X**のダイアグラムを**SDT 3.1X**以降のバージョンのエディタでファイルに保存すると、ダイアグラムは**3.1X**形式へ変換されます。また、ダイアグラムをオーガナイザにインポートすると、自動的に**3.1X**形式へ変換されます。[インポートされたダイアグラムを**SDT3**形式で保存 (*Save imported diagrams in SDT 3.X format*)]オプションは、このために使用します。

注意！

以前のバージョンのSDT（たとえば、SDT 3.0xやSDT 2.3）でSDT 3.1X形式のファイルを開いたり、以前のバージョンのMSCエディタで3.5の新しい形式のMSCダイアグラムを開くことはできません。ダイアグラムについては必ずバックアップコピーをとっておいてください。

生成したSDLシミュレータおよびエクスプローラ

シミュレータとエクスプローラは、これらを実行をサポートするSDTと同じバージョンのSDTで生成してください。異なるバージョンのSDLシミュレータGUIで実行すると、期待したとおりに動作しない場合があります。再生成するためには、[Full Make](#) ファシリティを使用して、すべてのCファイルが正しく再生成されるようにしてください。

SDLエディタのSDL/GRトレースが正常に機能するためには、SDLダイアグラムを保存する必要があります。（SDT 3.Xの参照メカニズムは、SDT 2.Xファイルに保存されていない情報が存在していると想定しています。）

配置ダイアグラム

バージョン4.2以降、配置ダイアグラムのファイル形式(*.sdp)が拡張されています。バージョン4.2の配置エディタでは、古いバージョンの配置エディタで作成した.sdpファイルを読み取って変換することができます。ただし、古いバージョンの配置エディタを使用して、バージョン4.2の配置エディタで作成されたダイアグラムを読み取ることはできません。

バージョン4.1で、配置ダイアグラムのファイル形式 (*.sdp) が拡張されました。バージョン4.1の配置エディタは、古いバージョンの配置エディタで作成された.sdpファイルを読み取って変換することができます。ただし、古いバージョンの配置エディタは、バージョン4.1で作成された配置ダイアグラムを読み取ることはできません。

バージョン4.1以降の配置エディタでは、ビルドスクリプトの生成はサポートされなくなりました。その代わりに、配置ダイアグラムを分割ダイアグラムモデルに変換する機能が追加されています。この機能は、ターゲティングエキスパートで使用されます。この機能により、ターゲティングエキスパートを使用してSDLシステムを構築する際に、配置ダイアグラムで分割設定を利用できるようになります。

ビルドスクリプトの生成がサポートされなくなったため、古い配置ダイアグラムを4.1形式に変換する場合は、ノードでのビルド設定とコンポーネントレベルは無視されます。オブジェクトレベルの限定子データは、自動的に変換されます。

TTCN Suite データベースの互換性

ここでは、さまざまなバージョンのTTCN Suiteデータベース形式と、そのデータベース形式間の変換方法について説明します。次の表は、TTCN Suiteの各バージョンで使用されている形式の一覧です。

| ITEXのバージョン | データベースのバージョン | 内容 |
|------------------------------|--------------|---|
| 2.0, 2.1 | 0 | ベースバージョン。 |
| 2.2 | 1 | このバージョンでは追加のアナライザ情報を保存します。 |
| 3.0x | 2 | TestCaseのDefaultsRefとConfigurationの順番を修正しました。 |
| 3.1x | 3 | モジュールのTTCNおよびエンコード/デコードのサポートを追加しました。構造とテーブルの両方を含む複合形式に変更しました。 |
| 3.2, 3.3, 3.4, 3.5, 3.6, 4.0 | 4 | アナライザ情報をいくつか追加で保存し、かつ、内容の構成を若干変更しました。 |
| 4.1, 4.2, 4.3, 4.4 | 5 | 解析データ構造を変更しました。 |
| 4.5 | 6 | 解析データ構造を変更しました。 |
| 4.5.4, 4.6 | 7 | 制約のグループがサポートされるようになりました。 |
| 4.6.1, 5.0, 5.1, 6.0, 6.1 | 8 | 解析データ構造を変更しました。 |

データベースのアップグレード

TTCN Suiteで古いデータベースを開くと、そのデータベースをTTCN SuiteのTTCN-MP形式で現行バージョンに変換できるかどうかをTTCNがチェックします。このとき、その古いバージョンのデータベースに対して変換プログラムを使用できなければなりません。このプログラムの名前には、データベースのバー

ジョン番号を示すサフィックスが付いています。たとえば、バージョン4のデータベースを変換するプログラムはmp-output-old4という名前です。

このプログラムは、**TTCN Suite 6.3**でインストールされます。バージョン4のデータベースをMP形式に変換するために使用されます。その結果作成されたMPファイルはGR形式に変換されます。この機能は、下位互換性問題を解決します。**TTCN Suite**または新しいバージョンの**TTCN Suite**は、古いバージョンの**TTCN Suite** データベースを開くことができます。

上位互換性

古いバージョンの**TTCN Suite**で新しいバージョンの**TTCN Suite**データベースを開く必要があるときは、特別な問題が発生します（上位互換性）。バージョン0、1、および2のデータベース間では転送形式（MP）が同じなので、この問題は発生しません。これに対し、バージョン3.1以降で使用されるMP形式には拡張子が付いています。また、バージョン3.1以降のデータベースが単一ファイル（複合形式）に保存されるときに、拡張子として、以前のバージョンのような

.itex/.itex-tablesの組み合わせではなく.itexが付くために、別の問題が発生します。

TTCN Suite 6.3では、以下のいずれかの型のドキュメントがデータベースに含まれます。

- テストスイート
- モジュールのテストスイート
- モジュール

モジュール型の新しいデータベースをバージョン3.0x以前の古いデータベースに変換することはできません。上位互換性がサポートされるのは、テストスイート型およびモジュールのテストスイート型のデータベースのみです。

この変換メカニズムを、現在インストールされている以前の3.Xのバージョンに追加することができます。このためには、インストール済みのバージョン6.3の、目的のアーキテクチャに対応したバイナリディレクトリから、mp-output-old6およびconvert-to-old6というプログラムをコピーして、バージョン3.Xのツリー内の対応するディレクトリに貼り付けます（主にこの用途のために、これらのファイルは現行バージョンに含まれています）。

バージョン6.3のデータベースからバージョン3.0xに変換できるようにするには、バージョン3.6のデータベースファイルだけでなく、拡張子.itex-tablesを持つもう1つのファイルが必要になります。このファイルは、何も含まれていないダミーファイルで十分です。

既知の問題

SDL and TTCN Suite ツール共通の既知の問題点

サポートされるアーキテクチャ

Linux と Sun Sparc 向けに配布される実行形式とオブジェクトライブラリは、32 ビットモードでコンパイルされています。

LP64 アーキテクチャ

サポート対象ではありませんが、LP64 アーキテクチャ（主として 64 ビット Solaris）上の SDL Suite について、生成コードに対する限定的なテストは行われています。Solaris については、64 ビットプラットフォーム向けに post.o の専用バージョン（lib/sunos5lib/post64.o）を提供しています。TTCN Suite は、LP64 アーキテクチャ上ではテストは行われていません。

一般的な問題点

一部のツール、シミュレータ/エクスプローラ、CIF ツール、および ASN.1 to SDL ツールなどは、空白を含むファイル名をサポートしません。

ASN.1 ユーティリティ

OBJECT IDENTIFIER の未定義のコンポーネントはレポートされません。

定義済みの OBJECT IDENTIFIER は SDL では値を持ちません。

PermittedAlphabet 制約を一部の文字列型に使用して、制約外の Tuple または Quadruple を割り当てると、エラーメッセージは表示されません。たとえば、`T ::= IA5String (FROM ("a" | "b")) t2` `T ::= { 1, 1 }` ではメッセージが生成されません。

32 ビット整数の範囲外の制約をもつ整数型の値は、正しく取り扱えません。

ASN.1 BER/PER エンコーディング/デコーディング

空の SEQUENCE は SDL では値を持ちません。以下の例を確認してください。

```
T1 ::= SEQUENCE {  
  v11 T1 ::= { }  
  this is mapped to the following SDL:  
  
  newtype T1  
  endnewtype;  
  synonym v11 T1 = ( . . );
```

SDLデータ型T1は値を持たず、SDLアナライザはこのsynonymを受け入れません。SDLインライン型に対してWITH COMPONENT制約を作成する際に、そのインライン型の情報は内部的なASN.1表記には保存されず、エンコーダ/デコーダ生成要求に対して型構造は生成されません。このため、SDL型にEncode/Decode操作が生成されると、SDLパッケージから生成されたCコードをコンパイルする際にコンパイル・エラーが発生します。これは存在しない型情報を参照しようとするためです。

ASN.1

```
T1 ::= SET OF INTEGER
T2 ::= T1 ( WITH COMPONENT( 0 .. 5 ) )
```

生成された SDL

```
newtype T1
  String (Integer, emptystring)
endnewtype;

syntype T2 = T2_INLINE_0 endsyntype;

newtype T2_INLINE_0 /*#SYNT*/
  String (T2_INLINE_1, emptystring)
endnewtype;

syntype T2_INLINE_1 = Integer constants 0 : 5
endsyntype;
```

T2_INLINE_0 と T2_INLINE_1 に関する情報は、SDLにのみ生成され、コーダー向けの型構造は生成されません。

簡単な回避方法として、WITH COMPONENT制約を使用せずに、SET OF型で新たな型を作成する方法があります。

```
ConstrainedINTEGER ::= INTEGER ( 0 .. 5 )
T1 ::= SET OF INTEGER
T2 ::= SET OF ConstrainedINTEGER
```

TTCN ジェネレータ: 自動タグ付けはTTCNに正しくマップされません。自動タグ付けを適用すると、すべての暗黙的な自動タグは明示的にTTCNにマッピングされます。これは、TTCN ASN.1 型に自動タグを定義する手段がないためです。ASN.1では、自動タグ付けは、IMPLICIT型のタグとみなされますが、TTCNに

マッピングする際にこの情報は失われ、タグはEXPLICITとして取り扱われ
ます。

```
M
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

S ::= SEQUENCE {
        x BIT STRING,
        y INTEGER
    }

END
```

フィールドxとyの型は、自動タグ付けの結果、[1] IMPLICIT および [2]
IMPLICITタグをもつことが期待されます。

TTCNへのマッピング

```
S ::= SEQUENCE {
        x [0] BIT STRING,
        y [1] INTEGER
    }
```

フィールドxおよびyのタグは、デフォルトでEXPLICITになります。これは
ASN.1のセマンティクスとは整合していません。

この問題は、タグがTTCN内部型記述からではなく各型の正しい生成型情報から
抽出されるため、コーディング/デコーディングには影響しません。

Java インターフェイス

コンパイルの前に、postmaster.javaファイルの53行目の、文字列"libdtpost"を
文字列"libsdtpost"に変更する必要があります。さらに、Windows上のライブラ
リーの名前を"libsdtpost.dll"から"sdtpost.dll"に変更する必要があります。

UNIXではこの作業は不要です。

オンライン ヘルプ

UNIX環境では、NETSCAPEをヘルプビューワーとして使用した場合、環境設
定マネージャのNetscapeCommandを使用できません。代替として、
FirefoxCommandを両方のビューワー用に使用します。また、ビューワーとして
Firefoxを選択している場合、環境設定マネージャの [デフォルトの設定] メ
ニューコマンドは灰色表示になりません。

Microsoft Wordとの統合機能(Windowsのみ)

MS Word文書内の二つのエンドポイント (endpoints) の間のリンクを追加、削除する場合、変更される専用文字形式はいずれか一方だけです。

印刷

UNIX環境でWebPrintオプションを使用にすると、使用されているすべての色が白色で表示されます。

テキスト・エディタ

UNIX環境でテキスト・エディタで文字を選択解除した場合、エンドポイントである選択対象下の下線の再描画ができません。

回避方法: Ctrl+L キーを使用すると、エンドポイントである文字の選択を選択解除した際に文字は再描画されます。

SDL Suiteの既知の問題

Cadvanced / Cbasic SDL to Cコンパイラ

コンポーネント選択の組み合わせによってはCのコンパイルエラーが発生します。以下のSDL型があるとします。

```
newtype s struct
  a, b integer;
endnewtype;
newtype t string(s, empty)
endnewtype;

dcl v t;
```

ここで、以下のように二段階のコンポーネント選択を行ったとします。

```
v(2) ! a
```

つまり、**string**コンポーネントの第二要素として構造体のコンポーネント**a**を選択すると、コンパイルエラーが発生する場合があります。これは、構造体コンポーネント選択は標準的なCの"."を使用して実行できますが、**string**コンポーネント選択は関数を使用して実行する必要があるからです。生成されるCコードは以下のようになります。

```
Extr_t(v, 2).a
```

問題は、一部のコンパイラが関数の結果上でのコンポーネント選択を受け入れないことです。

適切な回避方法は、コンポーネント選択を以下のように二段階に分けることです。

```
tmp := v(2),  
x := tmp ! a;
```

Cmicro SDL to C コンパイラ/Cmicro ライブラリ

SDL to C コンパイラは、SDL で定義された演算子のパラメータ リスト内に余分なマクロ呼び出しを生成します。これらのマクロ呼び出しは、コンパイルエラーを発生させます。これらはタイト インテグレーション時のみ影響をもちます。

XMK_USE_RECEIVER_PID_IN_SIGNAL を定義しない場合、いかなる場合も xRouteSignal() 関数を使用できません。

グラフィカル エディタでのコピーと貼り付け

1007394 (Windows のみ) : 日本語文字を含んだ文字列のコピー/貼り付けに問題があります。この問題は、二つのシンボル間で文字列をコピーする際のみに発生します。

回避方法 **Work-around**: 文字列をコピーした後、文字列をテキストエディタに貼り付け、その後テキストエディタからコピーし直して、第二のシンボルに貼り付けます。

CPP2SDL

UNIX 環境で CPP2SDL のプリプロセスのために CC または cc コンパイラを使用していて、プリプロセス中にメッセージを出力する場合、メッセージがプリプロセス結果のファイル内に出力されます。この結果、ファイル内容を SDL に翻訳する際に問題が発生します。

回避方法 **Work-around**: コマンドラインからファイルのプリプロセスを実行する際に、プリプロセス中にメッセージを出力しないようにします。

あるタグ付き値に関して、タグの名前が同じスコープにある別の宣言と重複している場合、そのタグ付き値は正しく取り扱われません。生成コードのコンパイル時にエラーが発生します。

例:

```
struct S {  
};  
int S;
```

回避方法: タグまたは宣言のいずれか一方の名前を変更します。

コードジェネレータは、参照を外す演算子 (*) を正しく取り扱いません。

コードジェネレータは、ポインタと多次元配列の複雑な組み合わせを正しく取り扱いません。

シミュレータは、仮想継承で基底クラスから継承している **public** メンバをもつ C++ クラスを使用した SDL システムをシミュレーションできません。ただし、アプリケーション生成は動作します。

Windows 環境で、CPP2SDL が構文エラーをヘッダファイルではなくプリプロセスされたファイルに報告することがあります。このようなエラーメッセージはオーガナイザログからナビゲートできません。

MSC エディタ

- 10149: [編集]、[スペースを空ける] コマンドを 101 以上または下図で選択しているシンボル数以上の値を指定して使用すると、オブジェクトが消失します。このエラーはダイアグラムが 1 ページのみで構成されている場合に発生します。オブジェクトを再取得するには、[編集]、[描画サイズ] I コマンドでダイアグラムのサイズを拡張します。
- 13737: MSC エディタをスタンドアローン（オーガナイザからの起動でない場合）で実行している場合、オーガナイザでの環境設定の変更は自動では MSC エディタには反映されません。変更を反映するには、**sdi.ini** ファイル（Windows 環境）または **.sdtpref** ファイル（UNIX の場合）を起動したエディタが参照しているディレクトリにコピーする必要があります。たとえば、**msctrace** ディレクトリのサンプルを使いたい場合は、環境設定ファイルを `<install dir>/examples/msctrace/<platform>` にコピーします。
- MSC エディタで MSC PR ファイルを開く際に、ダイアグラムが MSC GR に変換されていると PR に記述されたすべての注釈 (*/*...*/*) が消失します。
- シンボルと線が接続されなくなる場合があります。この問題に遭遇した場合、システムのクラッシュを避けるために以下の手順を行います。シンボルまたは線の末端をドラッグして再接続します。再接続できた場合は問題は発生しません。接続されなくなったシンボルや線が接続していたシンボルは絶対に消去しないでください。消去した後で接続されなくなったシンボルや線を移動するとエディタはクラッシュします。
- 複数ページにまたがるメッセージ・シーケンス図から EPS 形式ファイルを作成すると、最後のページの MSC のみを含む単一ファイルが生成されます。

SC to SDL コンバータ

SCからSDLへの変換は、SDLで大文字小文字の区別の設定がオフになっていると動作しません。

SDL エディタ

描画域外のテキスト:

1つのオブジェクトに大量のテキストを入力した場合、またはダイアグラムが最後に保存されたときに使用していたフォントよりもサイズの大きなフォントでダイアグラムを開いた場合、テキストが描画域からはみ出す可能性があります。この状況は、描画域全体が表示されている場合に描画域の右側や下側にテキストがあることによって検知できます。

テキスト エディタへの接続:

他のテキストエディタでオブジェクトを編集するために [テキストエディタへの接続] を選択して、テキストエディタを終了する前にオーガナイザを終了すると、大量のエラーメッセージが表示されます。

回避方法: オーガナイザを閉じる前に、テキストエディタまたはSDLエディタを閉じます。

SDLエディタでは、環境設定マネージャでのゲートシンボルのColor設定は無視されます。

SDL インデックス ビューワー

インデックス ビューワーの通常の検索機能で、一部のテキスト文字列を見つけられないことがあります。

回避方法: 検索開始前にシンボルが選択されていないことを確認してください。または代わりに高速検索機能を使用してください。

SDL 情報サーバー Info Server

SDL情報サーバーは、SDLエディタおよびタイプビューワーの信号辞書で使用されます。

信号辞書を使用する場合、SDL情報サーバーはtaskシンボル内のアルゴリズムから送信される信号を見つけられません。以下の例を参照してください。

```
task {if (IsCorrect)
      output ok;};
```

この信号"ok"は見つけられません。

以下の例は、SDL情報サーバーは解析できません。

```
newtype a inherits b
operators
putbits: W -> W;
endnewtype ;
```

以下の例は問題ありません。

```
newtype a inherits b adding
operators
putbits: W -> W;
endnewtype ;
```

SDL シミュレータ

グローバルプロシージャ内の遷移はカバレッジテーブルでは取り扱われません。

システムが外部シノニムを含む場合、"元に戻す"処理は動作しません。

Define-Continue-Mode がオンに設定されていると、"元に戻す"処理は動作しません。

SDL シミュレータ UI/エクスプローラ UI

UIでシミュレータまたはバリデータを行う場合で、実行中のシミュレータやエクスプローラをオーガナイザで再生成して [再開] を選択すると、UIがハングします。

文字列の外部にある引用符は、UI処理中に消費されます。したがって、SDLエクスプローラコマンド内でパターンを囲むにはアポストロフィを使用してください。

SDL エクスプローラ

グローバルプロシージャ内の遷移はカバレッジテーブルでは取り扱われません

SDLシステムでTimeシンタイプを使うと、xEq_SDL_Timeについてコンパイルエラーになります。

回避方法: Timeシンタイプは使用しないでください。

待ち行列がブロックインスタンス内のチャンネルまたはブロックインスタンスに対するチャンネルに追加されると誤動作が起り、Define-Queueコマンドが起動します。

式におけるANYは決定論的です。つまり、ただ1つの結果のみを与えます。

回避方法: 可能であれば、分岐のANYを使用してください。これによって、SDLエクスプローラはすべての分岐を探索できます。

シミュレータでは、入力が**none**のMSCトレースは入力なしの**none**の出力となります。このタイプのMSCはSDLエクスプローラでは検証できません。SDLエクスプローラでは、**none**入力のトレースはありません。

回避方法: このような入力を削除します。

ブロックインスタンスセットに対するMSC検証は正しく動作しません。SDLエクスプローラは、そのブロックタイプの特定のインスタンスに向けて外部から送信された信号を識別できません。

ターゲティング エキスパート

Windows 環境のみ only: 名前に空白を含むディレクトリパスにSDL Suiteをインストールすると、ターゲティングエキスパートは動作しません。

ターゲットディレクトリ構造 Target directory structure ...

SDLシステム (.sdt) ファイルが複数のSDLシステムを保持していてシステムの順番が変更された場合（たとえば、既存の2つのシステムの間に新しいシステムが挿入された場合など）に、ターゲットディレクトリ構造が正しく更新されません。

複数の異なるSDLシステム (.sdt) ファイルが1つのターゲットベースディレクトリをもつ場合に、ターゲットディレクトリが正しく設定されません。

コードライブラリを使ってOSE向けのThreaded Lightインテグレーションをビルドする際に、リンカのエラー（リンク時にefs_stderrが見つからない）が発生します。

回避方法: OSEライブラリがコードライブラリの後にリンクされるように、生成されたメイクファイルでリンクの順番を変更してください。

Threadedインテグレーションは、以下のオプションでは動作しません。MSCトレース、秒単位の時間の取り扱い、代替信号送信。

ターゲット テスタ

ターゲットライブラリについて [大きな信号パラメータを許可する] が設定されている場合、MSCトレースが誤動作して、大きなパラメータの指定された信号の出力が消失します。また、テキストトレースに以下のエラーメッセージが表示されます。

**ERR: Unknown trace message cannot be converted

**ERR: to MSCE message. Message will be ignored.

複合データ型の信号パラメータを使用すると、トレース内で誤った値が発生します。

ターゲットテストが、オーガナイザが実行中であるにもかかわらずオーガナイザとの接続を確立できないことがあります。ターゲットテストを再起動してください。二度目の試行で正しく動作します。

UNIX 環境のみ:ターゲットテストは、環境によっては/tmpディレクトリ下の一時ファイルを適切に削除できないことがあります。その結果、/tmpディレクトリに大量の.sdt.cmicro.*ファイルが残ることになります。手動でファイルを削除する (rm /tmp/.sdt.cmicro.*) か、Cmicroポストマスタからresetコマンドを使用 (\$sdtbin/sdtmpm -r) してください。この問題は、Solaris2.5のソケット接続の終了処理のバグに関連しています。

Threaded Light インテグレーション

TCP/IPモジュールを使用したOSE Threaded light インテグレーションをビルドする際に、ターゲットエキスパートのインクルードパスが設定されません。

回避方法: `-I$(OSEROOT)/sfk-solaris2/std-include` を、ターゲットエキスパートの [コンパイラ] タブの [インクルード] テキストボックスに追加してください。

TTCN Suite の既知の問題

一般的な問題点

Linux上のTTCN Suite 6.1は、WindowsまたはSolarisで作成された.itexファイル、またはLinux上のTTCN Suite 6.2またはそれ以上で作成された.itexファイルを開くことができません。この問題はバイナリファイル形式の無効なバイト順序(Linux TTCN Suite 6.1は誤ったバイト順序を使用しています)によります。したがって、Linux上のTTCN Suite 6.1で作成した.itexファイルがある場合、Linux上のTTCN Suite 6.1を使用して、MP形式に変換する必要があります。この変換を行わないと、WindowsおよびSolaris上のTTCN Suite、またはLinux上のTTCN Suite 6.2またはそれ以上を使って、.itexファイルを開くことはできません。

Windowsで3.1形式のテストスイートを開いたり、変換することはできません。これは、Windows向けのITEX 3.1バージョンが存在しないためです。Solaris上のITEX 3.1がある場合は、ファイルをMP形式にエクスポートして、その後Windowsにインポートできます。

Linuxプラットフォームでは、ロードできるテストスイートについて制限があります。バージョン4.6.2より前のバージョンのTTCN Suiteで保存されたitex形式のファイルは、Linux上のTTCN Suiteにはロードできません。回避策として、バージョン4.6.2で開いてMP形式に保存する、WindowsまたはSolaris環境で開いて保存する、などがあります。

Solarisでディレクトリリストが更新されない。

ファイル選択ダイアログでは、フィルタ コマンドを使用して、新しいフィルタを指定してファイルリストを更新できます。ただし、ディレクトリ リストは更新されません。つまり、新しいディレクトリを作成しても、そのディレクトリは表示されません。

MPファイル形式の問題

MPからあるテストスイートをインポートした場合、オーバービュー テーブルの情報 (Descriptionなど) の動的パート内のテーブルへの転送がうまくいかない可能性があります。

TTCN標準では、オプションでスイート識別子を最初に含まれたパス指定が許されています。このため、テストスイートがおなじ名前の上位のグループを含んでいる場合、パスの最初の部分がグループ識別子なのかテストスイート識別子なのか区別できなくなっています。

ITEXは、パスの最初の部分がスイート識別子と等しい場合、それはオプションのスイート識別子であると仮定し、インポートの際に取り除きます。MPにエクスポートする際には、常にすべてのパスの先頭に追加されます。したがって、ITEXは、常に、エクスポートしたMPファイルをインポートできます。ただし、ITEX2.0はスイート識別子を追加しないこと、そしてそのためにここで述べた問題がITEX2.0でエクスポートしたMPファイルをインポートする場合に発生することに注意してください。

残る問題は、パスに対するスイート識別子指定なしで、最上位グループ識別子と同じ識別子を持つテストスイートをインポートする場合です。この問題は、ITEXへのインポートの前に、テストスイートの名前をユニークな識別子に一時的に変更して、ITEXで再び名前を元に戻すことで回避できます。

オンライン ヘルプ

Splarisではビューアセクションに問題があります。

Solaris版では、オンラインヘルプの表示方法が2通りあります。ITEXリソースファイルで選択する方法と、設定マネージャで選択する方法です。この2つの設定

が異なっていると、どちらで設定したオンラインヘルプが使用されるかは、最初にどちらからオンラインヘルプを呼び出したかに依存します。

テーブル エディタ

タブ文字の取り扱い:

テーブルエディタでは、タブ文字を入力できません。ただし、MPファイルからのタブ文字は表示できます。この場合も、テーブルエディタでの幅表示と印刷時の幅表示が異なります。

TTCN アナライザ

4980: 送信または受信ステートメントの後に空の振り舞いラインがある場合、アナライザの分析が不完全になります。

Type lookup:

"type lookup"機構の実装の仕方によって、ルックアップする変数に到達する前に、その型が参照されるコンテキストが分かるようにする必要があります。下の例を見てください。

```
[{a 2} = variable_with_sequence_as_type]
```

このステートメントは、シーケンス値の型のルックアップに失敗します。

回避方法: 先に変数を参照するように書き換えてください。

```
[variable_with_sequence_type = {a 2}]
```

ITEX は、"Simple Type Definitions"の"Type Encoding"フィールドと"Structured Type Definition"の"Encoding Variation"フィールドを分析しません。これらのフィールドは、現在のITEXT実装では意味をもちません。

TTCN to C コンパイラ

6238: TTCNエンコーディング定義はCCGではサポートされません。これはエラーではなく制約です。

3002315: あるテストケース変数 "Foo" とTestCase "FooD"があった場合、コンパイルエラーが発生します。つまり、テストケースに、テストケース変数に"D"を追加した名前を付けることはできません。

SDL and TTCN Integrated Simulator

6556: TTCN シミュレータ カーネルは、OBJECT IDENTIFIER:をエンコードできません。

9122: Windows NTプラットフォームでは、シミュレータ起動時に TTCN シミュレータに対して手動で送信した信号は正しく処理されません。

10764: コンカレントTTCNを使用したSDL and TTCN Integrated Simulationでは、パラメータなしのCANCEL ステートメントは動作しません。

TTCN シミュレータの実行の中止:

SDL シミュレータ実行中にTTCN シミュレータを中止すると、それ以上の通信が停止します。TTCN シミュレータを中止した場合は、必ずSDL シミュレータを再起動してください。

GCI/EGCI API

1008933: GciCreate 関数は動作しません。実装の予定也没有ありません。

代替手段として、タスク (taskcreate、taskdeleteなど) の使用を推奨します。

第 2 章

リリースノート

この章では、**SDL Suite and TTCN Suite 6.3**に関する以下の項目について説明します。

- サポートされているコンポーネント
- 新しい機能と変更された機能

サポートされているコンポーネント

SDL Suite 6.3 and TTCN Suite 6.3には、以下のツールとドキュメントが収められています。

SDL Suite と TTCN Suiteで共有するツールと機能

- オーガナイザ
- 印刷ユーティリティ
- 環境設定マネージャ
- ポストマスタライブラリ
- ASN.1ユーティリティ

SDL Suiteツールと機能

- SDLエディタ
- CIFコンバータ
- Ispell インテグレーション
- Emacsとの統合機能 (UNIXのみ)
- MS Wordとの統合機能 (Windowsのみ)
- リンク マネージャ
- タイプ ビューワー
- SDLカバレッジ ビューワー
- インデックス ビューワー
- SDLアナライザ
- SDL Access (標準パッケージには含まれていません)
- CPP2SDLユーティリティ
- SDL to Cコンパイラ (Cbasic/Cadvanced/Cmicro/Cextreme)
- ターゲティング エキスパート
- SDL Cコンパイラ ドライバ (SCCD)
- シミュレーションライブラリとGUI
- バリデーションライブラリとGUI
- アプリケーションライブラリ
- マスタライブラリ
- Cmicroパッケージ
- スレッドインテグレーション
- ASN.1 BER/PERエンコード/デコード
- OMエディタ
- SCエディタ
- MSCエディタ

- HMSCエディタ
- 配置エディタ
- テキストエディタ
- UML2SDLユーティリティ
- OM Access

TTCN Suite ツールと機能

- ブラウザ
- ファインダ (Windows/Linuxのみ)
- テーブルエディタ
- TTCNリンク
- オートリンク
- データ辞書
- アナライザ
- TTCN to Cコンパイラ
- シミュレーションライブラリとGUI
- TTCN Access
- TTCNエクササイザ
- ASN.1 BER/PERエンコード/デコード
- 適用のためのTCP/IPモジュール

ユーザー用ドキュメント

- **Installation Guide**日本語版 – オンラインヘルプおよびインストールCDに収められたPDFファイル
- **Release Guide**日本語版 – オンラインヘルプとインストールCDに収められたPDFファイル
- **Getting Started**日本語版 (チュートリアル) – オンラインヘルプおよびインストールCDに収められたPDFファイル
- **Methodology Guidelines**日本語版 – オンラインヘルプおよびインストールCDに収められたPDFファイル
- **User's Manual** (一部日本語版) – オンラインヘルプとインストールCDに収められたPDFファイル

新しい機能と変更された機能

SDL Suite and TTCN Suite の 6.2 (6.2.0)以降、6.3までに追加または変更された機能を以下に示します。

SDL Suite and TTCN Suite ツールの共有機能

- 本リリースのSDL Suite and TTCN Suiteは、Visual Studio 2008 (Express versionを含む) コンパイラ環境でテストが行われました。
- 本リリースからツールの名称が、**IBM Rational SDL Suite**と**IBM Rational TTCN Suite**に変更されました。
従来**TTCN SDL Co-Simulator**と呼ばれていたシミュレーションツールの名称は、**SDL and TTCN Integrated Simulator**に変更されました。
- 前のリリース以降、解決された問題については、オーガナイザの [ヘルプ] メニューから「**Latest news**」を確認してください。

A
ASN.1 ユーティリティ
 禁止事項 13
ASN.1、ASN.1 での禁止事項 13
C
convert-to-old コマンド 49
M
mp-output-old コマンド 48, 49
MSC ダイアグラム、保存形式の互換性 46
MSC、Z.120 との互換性 5
S
SDL Suite and TTCN Suite に含まれるコンポーネント 64
SDL Suite and TTCN Suite に含まれるツール 64
SDL Suite and TTCN Suite の互換性の問題 1
SDL Suite and TTCN Suite の新機能 66
SDL to C コンパイラ、制限事項と禁止事項 23
SDL-96、SDL Suite の拡張機能 2
SDL ダイアグラム、保存形式の互換性 46
SDL、IBM Rational 独自の拡張機能 3
SDL、SDL Suite での禁止事項 7
SDL、SDL to C コンパイラでの禁止事項 23
SDL、Z.100 との互換性 2, 7
SDL、アナライザでの禁止事項 20
T
TTCN、TTCN Suite の禁止事項 11
TTCN、標準との互換性 11
X
X.680、ASN.1 ユーティリティでの禁止事項 13
Z
Z.100 の互換性 2
Z.100、SDL Suite での禁止事項 7
Z.100、SDL to C コンパイラでの禁止事項 23
Z.100、アナライザでの禁止事項 20
Z.105、ASN.1 ユーティリティでの禁止事項 13
Z.120 の互換性 5
あ
アナライザ、制限事項と禁止事項 20
え
エクスプローラ、生成したエクスプローラの互換性 47
お
、お問い合わせ情報 iv
か
カスタマ サポート iv
き
規則、表記（ドキュメントの） x
禁止事項
 ASN.1 ユーティリティ 13

SDL 7
SDL to C コンパイラ 23
SDL アナライザ 20
TTCN 11

さ
サポート iv

し
システム ファイル、保存形式の互換性 45
シミュレータ、生成したシミュレータの互換性 47
新機能 66

せ
説明書（ドキュメント） viii

と
ドキュメント構成 viii

ひ
表記規則（ドキュメントの） x

ふ
ファイル
保存形式の互換性 45
ファイル属性
.itex-tables 49

ほ
保存形式の互換性 45

ま
マニュアル（ドキュメント） viii

り
リリース ノート 63