

CA Business Service Insight

事前定義済みコンテンツガイド

8.2



このドキュメント(組み込みヘルプ システムおよび電子的に配布される資料を含む、以下「本ドキュメント」)は、お客様への情報提供のみを目的としたもので、日本 CA 株式会社(以下「CA」)により随時、変更または撤回されることがあります。

CA の事前の書面による承諾を受けずに本ドキュメントの全部または一部を複製、譲渡、開示、変更、複本することはできません。本ドキュメントは、CA が知的財産権を有する機密情報です。ユーザは本ドキュメントを開示したり、(i) 本ドキュメントが関係する CA ソフトウェアの使用について CA とユーザとの間で別途締結される契約または (ii) CA とユーザとの間で別途締結される機密保持契約により許可された目的以外に、本ドキュメントを使用することはできません。

上記にかかわらず、本ドキュメントで言及されている CA ソフトウェア製品のライセンスを受けたユーザは、社内でユーザおよび従業員が使用する場合に限り、当該ソフトウェアに関連する本ドキュメントのコピーを妥当な部数だけ作成できます。ただし CA のすべての著作権表示およびその説明を当該複製に添付することを条件とします。

本ドキュメントを印刷するまたはコピーを作成する上記の権利は、当該ソフトウェアのライセンスが完全に有効となっている期間内に限定されます。いかなる理由であれ、上記のライセンスが終了した場合には、お客様は本ドキュメントの全部または一部と、それらを複製したコピーのすべてを破棄したことを、CA に文書で証明する責任を負います。

準拠法により認められる限り、CA は本ドキュメントを現状有姿のまま提供し、商品性、特定の使用目的に対する適合性、他者の権利に対して侵害のないことについて、黙示の保証も含めいかなる保証もしません。また、本ドキュメントの使用に起因して、逸失利益、投資損失、業務の中断、営業権の喪失、情報の喪失等、いかなる損害(直接損害か間接損害かを問いません)が発生しても、CA はお客様または第三者に対し責任を負いません。CA がかかる損害の発生の可能性について事前に明示に通告されていた場合も同様とします。

本ドキュメントで参照されているすべてのソフトウェア製品の使用には、該当するライセンス契約が適用され、当該ライセンス契約はこの通知の条件によっていかなる変更も行われません。

本ドキュメントの制作者は CA です。

「制限された権利」のもとでの提供:アメリカ合衆国政府が使用、複製、開示する場合は、FAR Sections 12.212、52.227-14 及び 52.227-19(c)(1)及び(2)、ならびに DFARS Section 252.227-7014(b)(3) または、これらの後継の条項に規定される該当する制限に従うものとします。

Copyright © 2012 CA. All rights reserved. 本書に記載された全ての製品名、サービス名、商号およびロゴは各社のそれぞれの商標またはサービスマークです。

CA への連絡先

テクニカル サポートの詳細については、弊社テクニカル サポートの **Web** サイト (<http://www.ca.com/jp/support/>) をご覧ください。

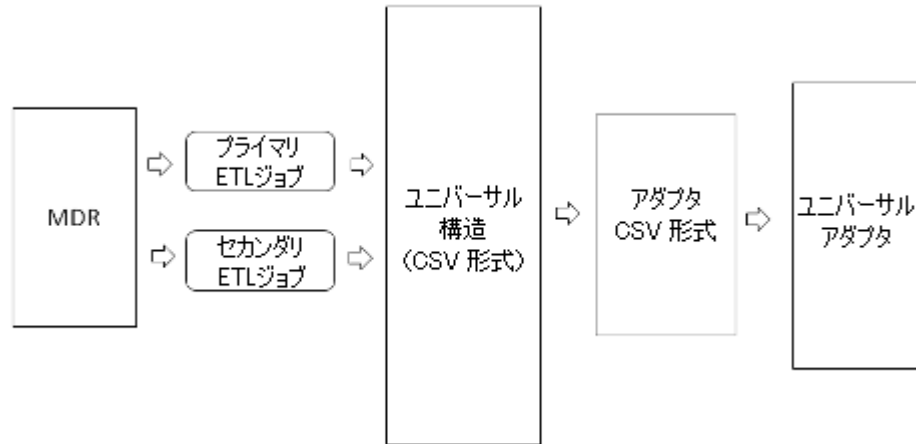
目次

第 1 章: 概要	9
対象読者	10
本ガイドの使用方法	10
ITIL プロセス モデリング ガイドライン	11
CA Business Service Insight および ITILv3 事前定義済みパッケージ	12
関数パッケージ	13
第 2 章: ITILv3 コンテンツ参照	15
Generic Samples テンプレート フォルダ	17
Generic Samples メトリック リスト	17
Service Delivery テンプレート フォルダ	18
Availability Management メトリック リスト	18
Capacity & Performance Management メトリック リスト	19
Service Support テンプレート フォルダ	21
Change Management メトリック リスト	22
Configuration Management メトリック リスト	23
Incident Management メトリック リスト	24
Incident Management – Multi Timeslot メトリック リスト	25
Problem Management メトリック リスト	26
Service Desk Management メトリック リスト	27
第 3 章: ITILv3 実装ノート	29
第 4 章: 関数パッケージ コンテンツ参照	31
Availability Management 関数リスト	31
Incident Management 関数リスト	32
Generic Formulas – Advanced 関数リスト	32
Generic Formulas – Elapsed Time 関数リスト	33

第 5 章: ビジネス ロジック モジュール	35
全般機能	37
Availability Management ビジネス ロジック モジュール	38
Availability and Failure Statistics モジュール	40
汎用ビジネス ロジック モジュール.....	42
Average Measurements モジュール.....	44
Count Unique モジュール	45
Elapsed Time モジュール	46
Generic Calculation モジュール	47
Success Calculations モジュール.....	49
Incident Management Business ビジネス ロジック モジュール	54
Percent Of Reopened Tickets Module モジュール	57
Resolution Time モジュール	59
Response Time モジュール	61
Timeslot Events Sender モジュール.....	63
Total Number of Incidents モジュール	64
Within X Business Days モジュール	65
Service Desk Management ビジネス ロジック モジュール.....	66
Call Center モジュール.....	69
Library モジュール.....	71
Resolution By Timeslots Extension Library	73
Within X Business Days By Timeslots Extension Library	74
Business Day Counting Library	75

第 6 章: 概要	77
第 7 章: インストールとアップグレード	79
第 8 章: その他の拡張機能	81
第 9 章: 既知の問題	83
第 10 章: 概要	85
第 11 章: インストールとアップグレード	87
第 12 章: その他の拡張機能	89
第 13 章: 既知の問題	91
索引	93

第 1 章：概要



このガイドでは、CA Business Service Insight の事前定義済みコンテンツ固有コンポーネント、テンプレートライブラリ構成、サービスレベル テンプレート、メトリック、関数、ビジネスロジックモジュール、およびイベント タイプを実際の契約とレポートの生成に使用する方法について説明します。

このガイドでは、以下の 2 種類のパッケージについて説明します。

- ITILv3 パッケージ
- 関数パッケージ

このセクションには、以下のトピックが含まれています。

[対象読者 \(P. 10\)](#)

[本ガイドの使用方法 \(P. 10\)](#)

[ITIL プロセスモデリング ガイドライン \(P. 11\)](#)

[CA Business Service Insight および ITILv3 事前定義済みパッケージ \(P. 12\)](#)

[関数パッケージ \(P. 13\)](#)

対象読者

このガイドは以下の読者を対象としています。

- サービスレベル アグリーメントの定義とシステムへの実装、サービスレベル テンプレートと契約テンプレートの作成、ビジネス ロジック 計算式とアダプタの実装を行っているサービスレベル マネージャ
- レポートの表示、生成、および保存と、キー コミットメントが保存されているかどうかダッシュボードを使用してモニタを行う必要がある担当者。
- アプリケーションのユーザと権限の管理、および地域オプションの定義、ログの管理、パッケージのインストールと解凍、システム基本設定の定義、データベースとライセンスの管理などの管理タスクを実行する必要がある管理者。

本ガイドの使用方法

本ガイドは以下のセクションに分割されています。

- [概要 \(P. 9\)](#) (このセクション) では、事前定義済みコンテンツの利点および ITILv3 事前定義済みコンテンツを使用して CA Business Service Insight がどのように ITIL 概念を満たしているかについて説明します。
- [ITILv3 コンテンツ参照 \(P. 15\)](#) では、コンテンツ内の主要コンポーネント、事前定義済みコンテンツ パッケージの構造、テンプレート ライブラリ構造、サービスレベル テンプレート、およびメトリックについて説明します。
- [ITILv3 実装ノート \(P. 29\)](#) では、実装方法と、事前定義済みコンテンツを使用して「実際の」契約およびレポートを作成する方法について説明します。契約をコミットしてレポートを生成することによって事前定義済みコンテンツ パッケージをロードするための、手順を追った説明を提供します。
- [関数パッケージコンテンツ参照 \(P. 31\)](#) では、事前定義済みコンテンツ パッケージで利用可能な関数の説明を提供します。
- [ビジネス ロジック モジュール \(P. 35\)](#) では、ビジネス ロジック モジュールおよびイベントタイプについて詳細な説明を提供します。

ITIL プロセス モデリング ガイドライン

ITIL (Information Technology Infrastructure Library) では、ビジネスの観点、サービスマネジメント、サポートおよび提供、Information and Communication Technology (ICT) インフラストラクチャ管理、およびアプリケーション管理を含む IT サービス マネジメントの主要な原則を規定しています。

- IT 運用ではビジネスの観点を理解し、ビジネス ニーズに応じた適切なサービス(サービス マネジメント)を提供することができるようにする必要があります。
- 次に、クライアントとの直接のやり取り(Service Delivery) およびサポート機能とは通常あまり関係のない、運用管理コンポーネントでサービスを構成します。
- IT アプリケーションを作成(または購入のち実装)し、保守します。
- これらすべてを ICT インフラストラクチャ上で実行します。
- IT セキュリティの管理および適切なシステム アクセスの提供は、ほかの原則のほとんどと重なります。

CA Business Service Insight および ITILv3 事前定義済みパッケージ

Information Technology Infrastructure Library (ITIL)は、情報技術(IT)インフラストラクチャの管理、開発および運用を行うための一連の概念およびポリシーです。

ITILは、高品質なITサービスの提供、およびITをサポートするために必要とされる施設と環境上の設備に関して指針を与える一連のガイドから構成されます

ITILは以下の5つの中心的なライフサイクルに基づいています。

- サービス戦略
- サービスデザイン
- サービストランジション
- サービスオペレーション
- 継続的サービス改善

ITILv3のガイダンスでは、以下の図に示すように、ライフサイクルの手法を採用しています。



CA Business Service Insight の事前定義済みコンテンツには、以下の主な利点があります。

- ベスト プラクティスおよび ITIL 標準に基づいたコンテンツを使用してより迅速な実装が可能
- キックオフから実稼働までにかかる工数および時間の著しい軽減
- ITIL プロセスの高速な同化の促進

関数パッケージ

関数パッケージには、視覚的なビジネス ロジック ワークスペースを使用して、実用的なクイック メトリックの定義を迅速にする関数のセットが含まれています。

パッケージには、関数と、可用性管理およびインシデント管理のドメインにフォーカスするビジネス ロジック モジュールが含まれます。また、汎用組み込み関数のライブラリに含まれる、重み付け平均などの汎用関数が用意されています。

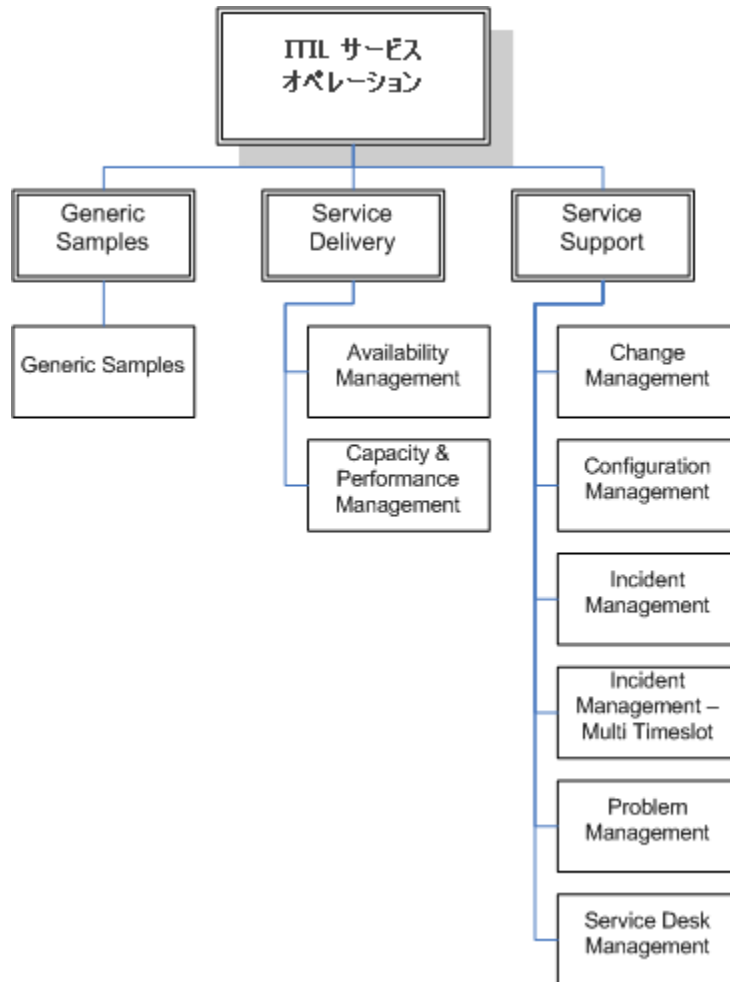
このパッケージを使用して、より高品質で標準化され、ITILv3 に沿った実装を、高速に実現することが可能です。一般的な使用例を以下に示します。

- Availability Management (可用性管理) - Percentage of Availability、Maximum Downtime、Number of Failures
- Incident Management (インシデント管理) - Percent Resolution on Time、Percent Response on Time、Average Response Time
- Generic Functions (汎用関数) - Weighted Average、Number of Events within Threshold、Average Elapsed Time

パッケージに含まれる関数は、CA Business Service Insight ITILv3 事前定義済みコンテンツ (個別のライセンスにて提供される) を活用するため、パッケージは汎用インストールの一部ではありません。CA Business Service Insight 環境へ個別にインポートする必要があります。

第 2 章: ITILv3 コンテンツ参照

事前定義済みコンテンツパッケージは、以下の図に示すように構成されています。



ITIL サービスオペレーションは、「通常どおりの業務」活動が行われる ITSM ライフサイクルの段階を示すテンプレートライブラリです。このライブラリ内のテンプレートフォルダでは、組織における日常業務とサービスの提供に使用されるインフラストラクチャが主な焦点となります。

ITIL サービスオペレーションは、以下のテンプレートフォルダで構成されています。

- **Generic Samples:** 以下のサービスレベル テンプレートが収容されています。

- **Generic Samples**

このサービスレベル テンプレートには、**Success Calculation** モジュールの使用方法を示すメトリックが含まれています。

- **Service Delivery:** 以下のサービスレベル テンプレートが収容されています。

- 可用性管理

- キャパシティおよびパフォーマンス管理

これらのサービスレベル テンプレートのいずれにも、IT サービス自体を管理する際に役立つメトリックが含まれています。

- **Service Support:** 以下のサービスレベル テンプレートが収容されています。

- 変更管理

- 構成管理

- インシデント管理

- インシデント管理 – 複数タイムスロット

- 問題管理

- **Service Desk** 管理

これらのサービスレベル テンプレートのいずれにも、IT サービスが提供されているかどうかを確認する際に役立つメトリックが含まれています。

このセクションには、以下のトピックが含まれています。

[Generic Samples テンプレートフォルダ](#) (P. 17)

[Service Delivery テンプレートフォルダ](#) (P. 18)

[Service Support テンプレートフォルダ](#) (P. 21)

Generic Samples テンプレート フォルダ

Generic Samples テンプレート フォルダには、以下のサービスレベル テンプレートが収容されています。

- Generic Samples

このサービスレベル テンプレートには、高度な非固有のメトリックの使用方を示すメトリックが含まれています。

メトリックのリストについては、「[Generic Samples メトリックリスト \(P. 17\)](#)」を参照してください。

Generic Samples メトリック リスト

以下の表に Generic Samples のメトリックを示します。

このサービスレベル テンプレートに関連したビジネス ロジック モジュールの詳細については、「汎用ビジネス ロジック モジュール (42P.)」を参照してください。

メトリック名	説明
Success Calculation Sample	関連するすべてのイベントで、成功したイベントの数を数えるか、または成功したイベントの割合を計算します。パラメータを使用して、無関係のイベントをフィルタリングしたり、重複するイベントやタイムスロット外部のイベントを無視したりできます。

Service Delivery テンプレート フォルダ

Service Delivery は IT サービス自体の管理です。IT サービスが、サービスプロバイダと顧客との間で合意したとおりに確実に提供されるようにする、多数の管理上のプラクティスが含まれています。このサービスレベル テンプレートに含まれるプロセスの本来の目的は、IT サービスの長期的な計画と改善にあります。

Service Delivery テンプレートフォルダには、以下のサービスレベル テンプレートが収容されています。

- 可用性管理

このサービスレベル テンプレートには、顧客とのサービスレベルレビューで使用する IT サービスの可用性のレベルを確実に特定できるメトリックが含まれています。その目標は、コスト効率よく、可用性レベルを保持して対象の相手へ期限内にサービスを提供する能力を最適化し、ビジネスにおいてビジネス目標を満たすことができるようにすることです。

メトリックのリストについては、「[可用性管理メトリックリスト \(P. 18\)](#)」を参照してください。

- キャパシティおよびパフォーマンス管理

このサービスレベル テンプレートには、絶え間なく変動するビジネス需要に応じてタイムリーかつコスト効率の良い方法で IT 処理とストレージ容量のプロビジョニングを確実に実施できるメトリックが含まれています。キャパシティ管理は、事業キャパシティ管理、サービスキャパシティ管理およびリソースキャパシティ管理として定義できます。

メトリックのリストについては、「[キャパシティおよびパフォーマンス管理メトリックリスト \(P. 19\)](#)」を参照してください。

Availability Management メトリック リスト

以下の表に Availability Management のメトリックを示します。

このサービスレベル テンプレートに関連したビジネス ロジック モジュールの詳細については、「[Availability Management ビジネス ロジック モジュール \(38P. \)](#)」を参照してください。

メトリック名	説明
Components number of failures (measured by monitoring tool)	計算期間内にサービスに失敗した回数をカウントします。この測定値を使用して、問題のある CI (構成アイテム) を特定して、置換または保持できます。

メトリック名	説明
Max single failure time	計算期間内の最大単一故障時間を測定します。
MTBF (Mean Time Between Failures)	サービスまたはサポートするコンポーネントが完全にリストアされたときから、同じサービスまたはコンポーネントの次の障害の発生までの平均経過時間。
MTBSI (Mean Time Between System Incidents)	現在実装されていません。
MTTR (Mean Time To Repair)	インシデントの発生からそのインシデントの解決までの平均経過時間。顧客サービスを修復するためにかかる平均時間。
Percent of service availability (based on monitoring tool events)	すべてのリソースの可用性に基づいた、合意済みサービス時間のうちの現在までの利用可能時間。サービスの現在のステータスを示します。
Percent of service availability (based on monitoring tool events) - Advanced	利用可能なリソースのしきい値に基づいた、合意済みサービス時間のうちの現在までの利用可能時間。サービスの現在のステータスを示します。
Percent of service availability (correlates monitoring tool events and incidents)	顧客によって報告されるインシデント管理システムのインシデントおよびモニタリング ツール イベントを測定します。
Percent of Service availability with direct impact on users (based on incidents)	顧客によって報告されるインシデント管理システムのインシデントを測定します。
Service number of customer impacting failures (correlates with incidents)	顧客によって報告されるインシデントとともにインシデント管理システムによって測定されます。
Service number of failures (measured by monitoring tool)	モニタリング ツール イベントに基づいて失敗の数を測定します。

Capacity & Performance Management メトリック リスト

以下の表に Capacity & Performance Management メトリックをリスト表示します。

このサービスレベル テンプレートに関連したビジネス ロジック モジュールの詳細については、「汎用ビジネス ロジック モジュール (42P.)」を参照してください。

メトリック名	説明
Average latency	平均遅延を計算します。

メトリック名	説明
Average throughput	平均スループットを計算します。
Customer satisfaction level	全体的な顧客満足度レベル。
Percent of CPU usage	CPU 使用率の割合を計算します。
Percent of file storage usage	ファイル ストレージの割合を計算します。
Percent of memory usage	メモリ使用量の割合を計算します。
Percent of time CPU exceeds utilization	CPU 使用率が事前定義済みしきい値を超える時間の割合。
Percent of time file storage exceeds utilization	ファイル使用率が事前定義済みしきい値を超える時間の割合。
Percent of time memory exceeds utilization	メモリの使用量が事前定義済みしきい値を超える時間の割合。
Percent of transactions completed on time	事前定義済みしきい値内で完了したトランザクションの割合。
Transaction average delivery time	トランザクションの平均デリバリ時間を計算します。
Worst transaction time	計算期間内の最悪トランザクション時間。

Service Support テンプレート フォルダ

Service Support テンプレート フォルダにより、IT サービスの提供が可能になります。また、IT サービスに対応した日常業務のプロセスも含まれています。

Service Support テンプレート フォルダには、以下のサービスレベル テンプレートが収容されています。

■ Change Management

このサービスレベル テンプレートには、計画および許可された方法で構成アイテムへのすべての変更が実施されているかどうかを検証するメトリックが含まれています。メンテナンスプロセスの変更、レビュー、優先度付け、計画、および実行のリクエストの作成に関連するインジケータがあります。

このサービスレベル テンプレートの目的は、制御対象となる IT インフラストラクチャへのすべての変更を迅速かつ効率よく処理するために標準化された方法と手順が使用されているかどうかを確認し、関連インシデントの数を減らし、これらのインシデントがサービスに及ぼす影響を最小限に抑えることです。

メトリックのリストについては、「[Change Management メトリックリスト \(P. 22\)](#)」を参照してください。

■ Configuration Management

このサービスレベル テンプレートには、サービスレベル テンプレートを管理し、これを(ビジネス)サービスと継続的に連携させることができるメトリックが含まれています。

このサービスレベル テンプレートを使用すると、サービスの提供に必要なすべてのリソースに関する情報をモニタおよび維持しながら、インフラストラクチャを制御することができます。単一のサーバのような単純なものから、IT 全体のような複雑なものまで、IT システム内の個別の構成アイテムがすべてトラッキングされます。

メトリックのリストについては、「[Configuration Management メトリックリスト \(P. 23\)](#)」を参照してください。

■ Incident Management

このサービスレベル テンプレートには、インシデントの記録、初期サポート、および分類とインシデントの解決に必要な調査および解決方法の決定に対応したメトリックが含まれています。このサービスレベル テンプレートの主な目的は、できるだけ迅速に正常なサービスオペレーションを復元し、業務活動への影響を最小限に抑えることです。

メトリックのリストについては、「[Incident Management メトリックリスト \(P. 24\)](#)」を参照してください。

■ Incident Management – Multi Timeslot

このサービスレベル テンプレートには、場所によってタイムスロットとタイムゾーンが異なる複数サイト環境でのインシデント管理に対応したメトリックが複数含まれています。

メトリックのリストについては、「[Incident Management – Multi Timeslot メトリックリスト \(P. 25\)](#)」を参照してください。

■ Problem Management

このサービスレベル テンプレートには、組織の IT サービスの正常な運用に影響するインシデントの解決と回避を確認するメトリックが含まれています。これには、障害が修正済みであることを確実にして、これらの障害の将来の再発を回避することが含まれます。また、予防処置メンテナンスのアプリケーションによって最初のインスタンスでこれらの障害が発生する可能性を縮小します。

メトリックのリストについては、「[Problem Management メトリックリスト \(P. 25\)](#)」を参照してください。

■ Service Desk Management (サービス デスク管理)

このサービスレベル テンプレートには、IT 部門へのすべての受信コールの単一の連絡先となるメトリックが含まれています。これにより、ユーザと IT 間に顧客中心のインターフェースが提供されるため、IT サービスの有効活用ができるようになります。

メトリックのリストについては、「[Service Desk Management メトリックリスト \(P. 27\)](#)」を参照してください。

Change Management [メトリックリスト](#)

以下の表に Change Management のメトリックを示します。

このサービスレベル テンプレートに関連したビジネス ロジック モジュールの詳細については、「[Incident Management ビジネス ロジック モジュール \(54P. \)](#)」を参照してください。

メトリック名	説明
Average change resolution time	変更プロセスにかかった時間の平均を測定します。プロセスの効率性を測定します。

メトリック名	説明
Number of incidents caused by improper implementation of change	現在実装されていません。
Percent of changes reopened	再オープンされた変更の割合を測定します。これらの測定値は Change Management の効率性レベルを反映します。
Percent of changes resolved on time	顧客期待度に従って解決された変更の割合。変更の成功の割合を測定します。
Percent of changes resolved on time, in business days	顧客期待度に従って解決された変更の割合 (営業日数で測定)。変更リクエストが行われた場所のタイムスロットおよびタイムゾーンに依存します。
Percent of changes resolved on time, on the same day	顧客期待度に従って、同日中に解決された変更の割合。変更リクエストが行われた場所のタイムスロットおよびタイムゾーンに依存します。
Percent of changes responded on time	顧客期待度に従って応答された変更の割合。
Percent of incidents escalated to changes	現在実装されていません。
Total number of changes closed	パイプラインでクローズされた変更の数。
Total number of changes in process	現在処理中の変更の数。
Total number of changes opened	パイプラインにある変更の数。 Change Management の作業負荷を測定します。

Configuration Management メトリック リスト

以下の表に **Configuration Management** のメトリックを示します。

このサービスレベル テンプレートに関連したビジネス ロジック モジュールの詳細については、「汎用ビジネス ロジック モジュール (42P.)」を参照してください。

メトリック名	説明
Number CI tracked	実際にログ記録およびトラッキングされた構成アイテムの数。
Number of incidents due to inaccurate CIs	不正確な構成情報によって発生したインシデントの数。

メトリック名	説明
Number of licenses not used	展開されていないソフトウェア ライセンスの数。 目的は未使用ライセンスのコストを最小化することです。
Percent of CMDB completeness	現在実装されていません。
Percent of inaccurate CIs	ログ記録された情報と比較して異なる構成を持つ構成アイテムの割合。

Incident Management メトリック リスト

以下の表に Incident Management のメトリックを示します。

このサービスレベル テンプレートに関連したビジネス ロジック モジュールの詳細については、「Incident Management ビジネス ロジック モジュール (54P.)」を参照してください。

メトリック名	説明
Average resolution time for priority X incidents	指定した優先度のインシデントの平均解決時間を測定します。
Average response time for priority X incidents	指定した優先度のインシデントに対する平均応答時間を測定します。
Max resolution time for priority X incidents	指定した優先度のインシデントの最大解決時間を測定します。
Max response time for priority X incidents	指定した優先度のインシデントの最大応答時間を測定します。
Min resolution time for priority X incidents	指定した優先度のインシデントの最小解決時間を測定します。
Min response time for priority X incidents	指定した優先度のインシデントの最小応答時間を測定します。
Percent of incidents escalated	現在実装されていません。
Percent of incidents reopened for priority X incidents	再オープンされたインシデントの割合を測定します。これらの測定値は Incident Management の効率性レベルを反映します。
Percent of priority X incidents resolved on time	契約によって各優先度に設定された目標解決時間に基づいて、インシデントの目標解決時間が達成された割合を測定します。
Percent of incidents resolved on time, in business days	契約によって設定された目標解決時間(営業日数)に基づいて、インシデントの目標解決時間が達成された割合を測定します。

メトリック名	説明
Percent of incidents resolved on time, on the same day	契約によって設定された目標解決時間(同日中)に基づいて、インシデントの目標解決時間が達成された割合を測定します。
Percent of priority X incidents responded on time	契約によって各優先度に設定された目標応答時間に基づいて、インシデントの目標応答時間が達成された割合を測定します。
Total number of incidents closed	計算期間内にクローズされたインシデントの数を数えます。これはヘルプデスクの効率性を測定します。
Total number of incidents in process	計算期間内に処理されたインシデントの数を数えます。これはヘルプデスクの効率性を測定します。
Total number of priority X incidents closed	計算期間内にクローズされたインシデントの数を数えます。これはヘルプデスクの効率性を測定します。
Total number of priority X incidents opened	計算期間内にオープンされたインシデントの数を数えます。これはヘルプデスクの効率性を測定します。

注: メトリック名内に X が表示されているものは、複数の優先度があることを意味します。

Incident Management – Multi Timeslot メトリック リスト

以下の表に Incident Management – Multi Timeslot のメトリックを示します。

このサービスレベル テンプレートに関連したビジネスロジック モジュールの詳細については、「Incident Management ビジネスロジック モジュール (54P.)」を参照してください。

メトリック名	説明
Location Timeslot Sender	メトリックのタイムゾーンおよびタイムスロットに基づいてタイムスロット入力およびタイムスロット終了イベントを送信します。
Multi timeslot percent of incidents resolved on time	顧客期待度に従って解決されたインシデントの割合。 変更リクエストが行われた場所のタイムスロットおよびタイムゾーンを区別します。
Multi timeslot percent of incidents resolved on time, in business days	契約によって設定された目標解決時間(営業日数)に基づいて、インシデントの目標解決時間が達成された割合を測定します。 変更リクエストが行われた場所のタイムスロットおよびタイムゾーンに依存します。

メトリック名	説明
Multi timeslot percent of incidents resolved on time, on the same day	契約によって設定された目標解決時間(同日中)に基づいて、インシデントの目標解決時間が達成された割合を測定します。 変更リクエストが行われた場所のタイムスロットおよびタイムゾーンに依存します。
Timeslot Events Receiver (Dummy Metric)	タイムスロットの入力および終了イベントタイプがパッケージの一部として含まれることを確認するための、タイムスロットの入力および終了イベントへの登録を含むダミーメトリックです。

Problem Management メトリックリスト

以下の表に Problem Management のメトリックを示します。

このサービスレベルテンプレートに関連したビジネスロジックモジュールの詳細については、「Incident Management ビジネスロジックモジュール (54P.)」を参照してください。

メトリック名	説明
Average problems resolution time	問題解決プロセスにかかった時間の平均を測定します。プロセスの効率性を測定します。
Percent of incidents escalated to problems	現在実装されていません。
Percent of problems reopened	再オープンされた問題の割合を測定します。これらの測定値は問題解決の効率性レベルに影響を与えます。
Percent of problems resolved on time	顧客期待度に従って解決された問題の割合。問題の成功の割合を測定します。
Percent of problems resolved on time, in business days	顧客期待度に従って解決された問題の割合(営業日数)。
Percent of problems resolved on time, on the same day	顧客期待度に従って、同日中に解決された問題の割合。
Total number of problems closed	パイプラインでクローズされた問題の数。
Total number of problems in process	現在処理中の問題の数
Total number of problems opened	パイプラインにある問題の数。Problem Management の作業負荷を測定します。

Service Desk Management メトリック リスト

以下の表に Service Desk Management のメトリックを示します。

このサービスレベル テンプレートに関連したビジネス ロジック モジュールの詳細については、「Service Desk Management ビジネス ロジック モジュール (65P.)」を参照してください。

メトリック名	説明
Average call duration	顧客の平均コール時間。
Average waiting time (for abandoned calls)	コールが放棄される前に顧客が待機した平均時間。
Average waiting time (for answered calls)	コールが応答されるまで顧客が待機した平均時間。
Percent of abandoned calls	合意済み待機時間の後に廃棄されたコールの数。
Percent of calls answered within a predefined number of seconds	コールの総数から指定した秒数内に応答されたコールの割合を計算します。
Percent of calls answered in more than 60s	60 秒の時間枠より後に応答されたコールの割合。
Percent of calls escalated to incident management	インシデントに変換されたコールのコール総数で割った割合。
Total number of calls	記録されたコールの総数。

第 3 章: ITILv3 実装ノート

実際の契約およびレポートを作成するために、CA Business Service Insight の事前定義済みコンテンツ パッケージを実装します。

実装ワークフローには以下の手順が含まれます。

1. 事前定義済みコンテンツ パッケージをロードします。
 - a. 事前定義済みコンテンツをアップロードします。
 - b. 事前定義済みコンテンツ パッケージを解凍します。

これらの手順の実行方法の詳細な説明は、「CA Business Service Insight ユーザ ガイド」の「パッケージ」セクションを参照してください。

2. 解凍した事前定義済みテンプレートライブラリをベースにしてカスタム テンプレートライブラリを構築します (顧客環境に合わせて微調整する必要のある仕様に焦点を絞る)。
 - a. 解凍した事前定義済みコンテンツ テンプレートライブラリをベースにしてカスタム テンプレートライブラリを作成します。
 - b. テンプレートフォルダを追加します。

注: ビジネス ニーズに最も合うようにテンプレートライブラリを調整できますが、オリジナルの ITIL サービス オペレーション テンプレートライブラリを変更しないことをお勧めします。これは、ユーザの ITILv3 構造を構築するための開始点です。

- a. サービスレベル テンプレートを追加します。
- b. メトリックの詳細を変更します。
- c. サービスレベル テンプレートをカスタマイズします。

これらの手順の実行方法の詳細については、「CA Business Service Insight ユーザ ガイド」の「テンプレートライブラリ」を参照してください。

3. 契約を作成します。
 - a. 作成したサービスレベル テンプレートに基づいて契約を作成します。これらの手順の実行方法の詳細については、「CA Business Service Insight ユーザ ガイド」の「テンプレートライブラリ」と「契約」を参照してください。
4. 事前定義済みコンテンツ イベントタイプを使用してデータを生成します。

- a. 新規契約をコミットする前に、受信データが事前定義済みイベントタイプと接続できることを確認します。イベントタイプは事前定義済みコンテンツに含まれていますが、アダプタは含まれていません。アダプタが含まれていない理由は、データソースが各顧客によって異なるということです。事前定義済みコンテンツと接続するためのアダプタを作成するとき、事前定義済みイベントタイプを選択し、入力に関連フィールドをイベントタイプフィールドへマップします。
- b. リソースモデルが登録と対応していることを確認します。カスタムのリソースタイプおよびリソースグループがある場合は、適切なリソースにそれらに関連付けて、それらを使用するすべてのメトリックの登録を更新する必要があります。事前定義済みコンテンツは、契約関係者およびサービスと関連付けられたリソースへの登録とともにデフォルトで含まれています。

これらの手順の実行方法の詳細な説明は、「CA Business Service Insight ユーザガイド」の「アダプタ」、「イベントタイプ」、および「リソース」のセクションを参照してください。

第 4 章: 関数パッケージ コンテンツ参照

CA Business Service Insight 事前定義済みコンテンツ パッケージには、CA Business Service Insight とともにリリースされる汎用関数に加えて追加の関数が含まれています。これにより、クイック メトリック モジュールからより多くの値を取得することができ、データを使用するレポート作成のエンド ツー エンド プロセスをボトムアップ アプローチで効率化します。

新しい関数は以下のサービスドメインに分割されます。

- Availability Management
- Incident Management
- Generic Formulas – Advanced
- Generic Formulas – Elapsed Time

注: 関数パッケージをインポートするには以下を実行します。

目的のフォルダへ手動でまたはパッケージ機能のアップロード機能を使用してパッケージをアップロードします。アップロード後、インポート ページ (パッケージ ページではなく) でのみパッケージが利用可能として表示されます。

インポート機能を使用して、パッケージをインポートします。

このセクションには、以下のトピックが含まれています。

[Availability Management 関数リスト \(P. 31\)](#)

[Incident Management 関数リスト \(P. 32\)](#)

[Generic Formulas – Advanced 関数リスト \(P. 32\)](#)

[Generic Formulas – Elapsed Time 関数リスト \(P. 33\)](#)

Availability Management 関数リスト

以下の表に Availability Management の関数を示します。

関数名	説明
Percentage of Availability	すべてのコンポーネント(リソース)が利用可能な時間の割合を計算します。計算はアップ/ダウン イベントに基づいて実行されます。

関数名	説明
Maximum Downtime	期間内に発生した最大のダウンタイムを計算します。計算はアップ/ダウン イベントに基づいて実行されます。
MTBF (Mean Time between Failures)	失敗と失敗の間の平均時間を計算します。計算はアップ/ダウン イベントに基づいて実行されます。
Number of Failures	期間内の失敗の数を計算します。計算はアップ/ダウン イベントに基づいて実行されます。
MTTR (Mean Time To Repair)	インシデントの発生からそのインシデントの解決までの平均経過時間を計算します。顧客サービスを修復するためにかかる平均時間。計算はアップ/ダウン イベントに基づいて実行されます。

Incident Management 関数リスト

以下の表に Incident Management の関数を示します。

関数名	説明
Percent Resolution on Time	期限内に解決されたチケットの割合を計算します。
Average Resolution Time	平均チケット解決時間を計算します。
Maximum Resolution Time	チケットの解決にかかった最大時間を計算します。
Minimum Resolution Time	チケットの解決にかかった最小時間を計算します。
Percent Response on Time	期限内に応答されたチケットの割合を計算します。
Average Response Time	平均チケット応答時間を計算します。
Maximum Response Time	最大チケット応答時間を計算します。
Minimum Response Time	最小チケット応答時間を計算します。

Generic Formulas – Advanced 関数リスト

以下の表に Generic Formulas - Advanced 関数を示します。

関数名	説明
Weighted Average	選択したフィールドおよびウェイトの重み付け平均値を計算します。
Number of events within Threshold	選択した値が特定のしきい値内にあるイベントの数を数えます。

関数名	説明
% of events within Threshold	選択した値が特定のしきい値内にあるイベントの割合を計算します。
Percentage of value	値の別の値に対する割合を計算します。
Count Unique	選択したフィールドで一意的な値を持つイベントの数を数えます。

Generic Formulas – Elapsed Time 関数リスト

以下の表に Generic Formulas - Elapsed Time 関数を示します。

関数名	説明
Average Elapsed Time	2 つの日付フィールド間の平均経過時間を計算します。
Maximum Elapsed Time	2 つの日付フィールド間の最大経過時間を計算します。
Minimum Elapsed Time	2 つの日付フィールド間の最小経過時間を計算します。
Percentage of Elapsed Time within Threshold	2 つの日付フィールド間の経過時間がしきい値内にあるイベントの割合を計算します。

第 5 章: ビジネス ロジック モジュール

以下のビジネス ロジック モジュールが事前定義済みコンテンツ パッケージとともに提供されます。

ドメイン	モジュール名	説明
Availability Management (可用性管理)	Availability and Failure Statistics	サービスまたはコンポーネントの可用性を計算します。さらに、可用性およびエラーの統計計算結果を示します。
	Average Measurements	単純平均および加重平均を計算します。
	Generic Calculation	最小値、最大値、合計値、パーセンテージなどのような一般的な計算の結果を示します。
Generic Modules (汎用モジュール)	Success Calculation	ユーザ定義の成功基準を使用して成功イベント数を算出します。
	Percent of Reopened Tickets	計算期間内での再オープンされたチケット数を計算します。
	Resolution Time	インシデントの解決時間は、インシデントがオープンされた時点から最終解決策が提供されるまでの経過時間として算出されます。
	Response Time	応答時間は、インシデントがオープンされてから、そのインシデントに対して最初の処理が行われるまでの経過時間として計算されます。
	Timeslot Events Sender	メトリックのタイムゾーンおよびタイムスロットに基づいてタイムスロットの開始および終了イベントを送信します。これにより、タイムスロットの影響を受けるメトリックにロケーション固有のタイムスロット情報を提供します。
Total Number of Incidents	計算期間内に受信されたインシデントをカウントします。	

ドメイン	モジュール名	説明
Service Desk Management	Call Center	コール センターのパフォーマンスの統計情報を算出します。

このセクションには、以下のトピックが含まれています。

[全般機能](#) (P. 37)

[Availability Management ビジネス ロジック モジュール](#) (P. 38)

[汎用ビジネスロジック モジュール](#) (P. 42)

[Incident Management Business ビジネス ロジック モジュール](#) (P. 54)

[Service Desk Management ビジネスロジック モジュール](#) (P. 66)

[Library モジュール](#) (P. 71)

全般機能

すべてのモジュールで共通して使用される全般的な機能を以下に示します。

■ デバッグ

メトリックで **Debug** という名前のパラメータが提供されており、その値が **Yes** である場合、モジュールはメッセージをログに書き込みます。メッセージは、各イベントハンドラ、および必要に応じてモジュール内のさまざまな箇所に書き込まれます。

■ コールバック

Success Calculations モジュールでは、コールバックを使用することでモジュールの基本機能の拡張を可能にします。コールバックは以下の箇所でサポートされます。

- 期間の開始前
- 期間の開始後
- タイムスロット開始前
- タイムスロット開始後
- イベントの前
- イベントの後
- ロード前
- ロード後
- 結果算出前
- 結果算出後
- 期間終了
- メトリック変更時
- タイムスロット終了時
- 登録時

イベントの前にコールバックが実行されると **true** が返されます。この場合、そのイベントハンドラの残り実行箇所はスキップされます。これにより、ユーザがイベントハンドラの機能を独自に作成した機能で置き換えることができます。

結果算出前のコールバックでは、**Result** に対する別の値がコールバック関数にセットされます。

結果算出後のコールバックでは、算出された結果(パラメータとして取得)がすでにセットされているので、その情報がコールバック中で使用できます。

Availability Management ビジネス ロジック モジュール

以下のセクションでは、可用性および失敗の統計情報に共通のメトリックで使用されるビジネス ロジック モジュールについて説明します。

可用性管理での前提

Availability ドメイン以下のすべてのモジュールでは以下の仮定および基本動作が前提となります。

- このコンポーネントについてレポートするイベントを受信する前の、最初の計算期間でのコンポーネントの初期ステータスは「Up」です。
- NONE というステータスがあります。リソース ステータスが不明で、NONE ステータスになっている場合は、リソースの前回までの実際のステータスに基づいてステータスを決定します。考慮します。
- タイムスロット外の期間が定義されている場合は、常に例外期間も同様に処理されます。
- 計算期間全体がタイムスロット外に該当する場合、期間の計算結果は NULL です。
- 強制イベント: 監視ツールによるサービス可用性についての情報が正しくないかまたは有効でない場合。このような場合には追加のイベントを定義し、それ以前に受け取ったすべてのイベントのステータスを強制的に追加のイベントのステータスに変更します。たとえば、サービスのステータスが Up であるとレポートされ、Force Down イベントを受信した場合、サービスは Down と見なされます。
- インシデント イベント: 可用性ステータスに関するインシデント イベントレポートでの場合。たとえば、優先度 1 チケットでの「インシデント - オープン」は DOWN ステータスを表わします。一方、優先度 1 チケットの「インシデント - 解決済み」は、UP ステータスを表わします。
- 新しいイベントを受け取ると、計算式が以前のイベントのステータスを確認します。前のイベントのステータスが Down であれば、このイベントと新しいイベントの間の期間は Down と見なされます。

- イベントは、特定のコンポーネントまたはデバイスのステータスについてレポートされるか、またはサービス全体についてレポートされる場合があります。全体イベントとコンポーネント イベントの両方がレポートされた場合は、全体のものが実際のステータスと見なされます。

Availability Management イベントタイプ

以下の Availability Management イベントタイプにより、特定のデバイスまたはサービスの可用性のステータスが提供されます。

- Availability UP イベント
- Availability DOWN イベント
- Availability Force UP イベント
- Availability Force DOWN イベント
- Availability Force NONE イベント

上記のものに加えて、インシデント イベント(「 Incident Management ビジネス ロジック モジュール」で説明)を使用してサービスまたはデバイスの可用性ステータスが反映される場合もあります。たとえば「Incident – Open」は DOWN ステータスを表し、「Incident – Resolved」は UP ステータスを表します。

Availability イベントタイプの構成

以下の表にすべての Availability イベントタイプの構成を示します。

#	名前	フィールドタイプ	フィールドの説明	計算に必要
1	Component	文字列	可用性のレポート元のリソース。	いいえ
2	AvailabilityVal	浮動小数点数	可用性のパーセンテージ	いいえ

Availability and Failure Statistics モジュール

サービス/コンポーネント可用性のパーセンテージでの前提

サービス/コンポーネントがある期間内に利用可能であった時間のパーセンテージを計算します。

可用性の計算は個別のコンポーネントまたはシステムのいずれかに対して行われます。可用性ビジネスロジックの実装例については、「実装ガイド」の「付録 2」で「ケーススタディ 14: 時間集積クロックの使用法」を参照してください。

可用性は以下のような方法で計算されます。

- 各計算期間についての計算方法

サービスまたはコンポーネントの可用性は、合計 AST (合意済みサービス時間: 計算期間の開始時点から終了時点までの時間からタイムスロット外期間を除いた時間として計算される) から AST 内の実際の休止時間 (タイムスロット外期間を除く計算期間内でのすべての休止期間の合計として計算される) を引いて求められた値を合計 AST で割り、その結果に 100% を掛けて算出されます。

リソースクラスタ外に構築されたシステムの可用性

- システムロジックに基づいて、ある時点でシステムが利用可能かどうかを決定します。システムロジックはクラスタ内のリソース間での負荷分散のタイプによって決まります。
- コンポーネントの負荷分散: Threshold パラメータで定義されたアイテム数以上が UP であれば、システムは UP になります。

失敗統計情報での前提

失敗統計情報が作成され、Availability ドメインでのいくつかのタイプの計算結果が配布されます。これらの計算はサービス/コンポーネントの可用性のパーセンテージと似た性質のものであるため、同じモジュール内に実装されています。モジュールは、WhatToCalculate パラメータによって決定される計算の結果を提供します。

- 休止時間が複数回報告された場合でも、1 回の失敗としてカウントされます。イベントが重複している場合は、休止時間の初回開始時がカウントされません。
- 休止時間は、それがタイムスロット期間内の合意済みサービス時間に該当する場合は、失敗と見なされます。

モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
WhatToCalculate	計算結果を決定します。
しきい値	システムが UP であるために UP になっている必要があるリソースの数(またはパーセンテージ)。
ThresholdUnit	しきい値の単位(アイテム数またはパーセンテージ)。
Debug	デバッグ モードであるかどうかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Parameter Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

- MAX - 計算期間内でシステムが利用不可能だった最大期間。
- MTBF - 失敗間平均時間。計算期間中での失敗間の平均時間。
- MTTR - 復旧までの平均時間。失敗の発生から解決までに経過した時間の平均。
- NumOfFailures - 計算期間内でシステムが利用不可能になった回数。
- Availability - 計算期間内でシステムが利用可能だった時間のパーセンテージ。
- PercentOfDowntime - 計算期間内でシステムが利用不可能だった時間のパーセンテージ。
- AvailabilityTime - 計算期間内でシステムが利用可能だった最大期間。
- AvailabilityTime - 計算期間内でシステムが利用不可能だった最大期間。

汎用ビジネス ロジック モジュール

このセクションでは、いくつかのドメインの計算や、キャパシティ/パフォーマンス管理、設定管理、顧客満足などの計算タイプで利用できる汎用のビジネスロジックモジュールについて説明します。

設定管理のメトリックはすべて **Generic Calculation** モジュールを利用しています。

キャパシティ/パフォーマンス管理のメトリックおよび顧客満足のメトリックは平均測定モジュールを利用しています。

汎用モジュールの入力用に使用されるイベントタイプの例を以下に示します。

Performance イベントタイプの構成

#	名前	フィールドタイプ	フィールドの説明
1	Timestamp	日付	測定時刻。
2	Component	文字列	パフォーマンスのレポート元のリソース。
3	CPU	文字列	% of CPU load.
4	Memory	浮動小数点数	使用メモリのパーセンテージ。
5	Throughput	浮動小数点数	バイト/秒の平均値。

6	Latency	浮動小数点数	平均秒数。
---	---------	--------	-------

Transaction イベントタイプの構成

#	名前	フィールドタイプ	フィールドの説明
1	Timestamp	日付	測定時間。
2	Transaction	文字列	実行されるトランザクションの ID/タイプ。
3	Component	文字列	パフォーマンスのレポート元のリソース。
4	Transaction Description	文字列	トランザクションに関する追加情報。
5	Duration	浮動小数点数	トランザクションの応答時間。

Customer Satisfaction イベントタイプの構成

#	名前	フィールドタイプ	フィールドの説明
1	Customer	文字列	顧客名。
2	SatisfactionLev	整数	顧客満足レベル。

Configuration イベントタイプの構成

#	名前	フィールドタイプ	フィールドの説明
1	Customer	文字列	顧客名。
2	NumberOfCIs	整数	構成アイテム数。
3	NumOfIncidents	整数	アセットの不適切な設定によって引き起こされたインシデントの数。
4	NumOfDeviations	整数	設定リポジトリおよび実際のアセット構成の間で特定された偏差の数。
5	NumOfUnAcctLic	整数	購入されたがリポジトリ内にはないライセンスの数。

Average Measurements モジュール

Average Measurements モジュールには、標準的な平均計算の計算式が含まれています。カスタムのビジネス ロジック モジュールで活用することを想定して用意されています。

Average Measurements モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
Valuefield	計算に必要なイベントフィールドの整数値を指定します。
WeightField (int)	数値は、計算に含めるウェイト値がどのイベント フィールドに含まれるかを示します。
Debug	デバッグ モードであるかどうかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Parameter Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

Average Measurements モジュールを使用するメトリックは単純平均または加重平均の計算を行います (関連する値フィールドがパラメータとしてレポートされた場合)。そのため、WhatToCalculate パラメータはありません。

イベント登録

数値フィールドを含む任意のイベントが登録に使用できます。

Count Unique モジュール

選択されたフィールドでの一意の値を持つイベントをカウントします。

Count Unique モジュールのインターフェース

パラメータを以下の表に示します。

名前	説明
ValueField	一意イベントを確認するイベントフィールドの名前。
Debug	デバッグ モードであるかどうかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Parameter Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Entity Counter Library (ECL)	エンティティカウンタの追跡に使用するライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

Count Unique モジュールを使用するメトリックは、一意イベントを単純計算するものであるため、WhatToCalculate パラメータはありません。

イベント登録

一意イベントをカウントする際のベースとなる、文字列フィールドを持つ任意のイベントが登録に使用できます。

Elapsed Time モジュール

2 つの時間フィールド間の経過時間を計算するための計算セットを提供します。タイムスロット期間外は計算から除外されます。

Elapsed Time モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
WhatToCalculate	計算期間の計算結果を定義します。
TimeUnit	時間の計算結果の表示単位 (秒、分、時間または日数) を定義します。
TargetTime	各イベントの比較を行う範囲を示すしきい値です。このしきい値内のイベントのパーセンテージが計算されます。
Debug	デバッグ モードであるかどうかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Parameter Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Aggregation Library (AL)	一般的な計算の実行に使用されるライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

- %WITHINTHRESHOLD/PERCENT - 2 つの日付フィールド間の経過時間がしきい値未満であるイベントのパーセンテージです。
- CNTWITHINTHRESHOLD - 2 つの日付フィールド間の経過時間がしきい値未満であるイベントの数です。
- AVG - 平均経過時間
- MAX - 最長経過時間
- MIN - 最短経過時間

イベント登録

2 つの日付フィールド (Date1、Date2) を持つ任意のイベントが登録に使用できます。

Generic Calculation モジュール

Generic Calculation モジュールは、フィールドへの入力値の最小値、最大値、パーセンテージなどの一般的な計算機能を提供します。

Generic Calculation モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
WhatToCalculate	計算期間の計算結果を定義します。
Valuefield	計算に必要なイベントフィールドのシーケンス値を整数で指定します。
Valuefield2	計算に必要なイベントフィールドのシーケンス値を整数で指定します。 WhatToCalculate が ADVANCEDPERCENT に設定されている場合にのみ使用されます。
Debug	デバッグ モードであるかどうかを決定します。
Threshold (int)	計算のしきい値を指定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Parameter Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Aggregation Library (AL)	一般的な計算の実行に使用されるライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

- **AdvancedPercent** - 2 番目の値フィールドの合計をもとに、1 番目の値フィールドの合計のパーセンテージを計算します。
- **%WITHINTHRESHOLD/PERCENT - ValueField** パラメータで指定されているフィールドがしきい値未満であるイベントのパーセンテージを計算します。
- **%CNTWITHINTHRESHOLD - ValueField** パラメータで指定されているフィールドがしきい値未満であるイベントをカウントします。
- **AVG - ValueField** パラメータで指定されているフィールドの平均値です。
- **MAX - ValueField** パラメータで指定されているフィールドの最大値です。
- **MIN - ValueField** パラメータで指定されているフィールドの最小値です。
- **COUNT** - イベント数をカウントします。
- **SUM - ValueField** パラメータで指定されているフィールドの合計値です。

イベント登録

数値フィールドを含む任意のイベントが登録に使用できます。

Success Calculations モジュール

このモジュールはイベントを受け取り、受け取ったイベントを成功基準を使用して確認します。以下の計算を実行できます。

- **Count** - トラッキング期間中の成功基準を満たしているイベントの数をカウントします。
- **Percent-Of Success** - トラッキング期間中に受信したすべてのイベントのうち、成功基準を満たしているイベントの割合を算出します。

使用例

サポートコールを処理するイベントタイプが存在し、フィールドの1つがコール期間を示しています。このモジュールを使用して30分以上に続いたコールのパーセンテージを計算することができます。

全般機能

Success Calculations モジュールの一般的な機能を以下に示します。

- パラメータに基づく柔軟な動作

Success Calculations モジュールには、動作設定に使用されるパラメータのセットがあります。

すべてのパラメータにはデフォルト値があり、パラメータがメトリック内に定義されていない場合、このデフォルト値が使われます。これにより、モジュールでサポートされているパラメータすべてについての設定を作成しなくても、デフォルトの設定で動作する簡易的なメトリックを作成することができます。

- 簡易的なパラメータ設定と詳細なパラメータ設定

ユーザは簡易的な比較方法を定義することができます。そのような比較方法では不十分で、モジュールの柔軟性を最大限に利用したい場合は、複雑な評価式を使用してモジュールの動作を設定することもできます。

- フィルタリング

Success Calculations モジュールではフィルタリングがサポートされます。これには **FilteringCriteria** という名前のオプションのテーブルパラメータを使用します。このパラメータでユーザは何らかの基準コード(たとえばフィールド X が 3 と等しいなど)を記述することが可能で、この基準に満たないイベントはすべて除外され、モジュールの処理対象になりません。テーブルパラメータの詳細な構成は「[基準パラメータ構成 \(P. 51\)](#)」に説明されています。メトリックでこのパラメータが指定されていない場合は、すべての受信イベントが処理されます。

- 一意識別

Success Calculations モジュールでは処理対象を一意に識別することができます。これは、メトリックがフィールド ID を含むパラメータを提供することができることを意味します。モジュールがフィールド名の値が同一であるイベントを同一期間中に受け取った場合、それらのイベントの 1 番目のもののみが処理されます。このパラメータを指定しないか、パラメータに何も指定されていない場合、イベントはすべて処理されます。

- タイムスロットの考慮

Success Calculations モジュールは、パラメータの設定値に応じてタイムスロットを考慮または無視することができます。これには **IgnoreTimeslots** という名前のオプションのパラメータを使用します。このパラメータの値が **Yes** であれば、モジュールはタイムスロット期間外に受け取ったイベントをすべて無視します。このパラメータが **No** であるかパラメータが定義されていなければ、メトリックはイベントをすべて処理します。

基準パラメータの構成

Success Calculations モジュールでは、フィルタ基準および成功基準という、基準を定義する 2 つのパラメータがあります。これらのパラメータの構成は同一です。

現在のイベント詳細(イベントハンドラからの `eventDetails` パラメータ)を使用して関数が呼び出され、処理中のイベントの詳細について成功テストが実行されます。このテーブルパラメータには、現在のイベントを成功と見なすかどうかを判断するのに必要な情報がすべて含まれています。

戻り値は、計算結果に基づいて `true` または `false` のいずれかが返されます。

このテーブル内にエントリが存在しない場合は、エントリがすべて成功であると見なされることとなります。

成功基準は単一または複数のフィールドを特定の値と比較するか、または評価式との一致を評価することで判定されます。複数の基準行は、ユーザが定義する論理関係に基づいて組み合わせられます。

たとえば、フィールドの結果が 3 から 6 の間であるイベントを成功イベントと定義している場合、2 つの基準行を `and` 論理演算子で組み合わせると以下のように問い合わせが行われます。

```
result > 3 and result < 6
```

テーブルの形式を以下に示します。

行番号	引数 1	演算子	引数 2	行の論理関係
1				
2				
...				

テーブルのフィールドは以下のとおりです。

- 引数 1 - このフィールドには、値を取り出すイベント詳細内のフィールド名を入力します。**Eval** オペレータが使用される場合には、このフィールドは無視されるので、空のままで構いません。また、このフィールドには以下を含めることができます。
 - **@Resource** - イベントのリソース名 (`eventDetails.Resource`) を返します。
 - **@EventType** - イベントのイベントタイプ (`eventDetails.EventType`) を返します。
 - **@SenderMetric** - 処理対象のイベント再利用イベントを送信したメトリックの詳細 (`eventDetails.Sender`) を返します。
- オペレータ - このフィールドには、引数 1 および引数 2 を比較する演算子を入力します。このフィールドには以下のいずれかの値を使用できます。
 - <
 - >
 - <=
 - >=
 - <> (等しくない)
 - **Eval**

通常の比較演算子のいずれかを使用した場合、システムは選択された比較演算子を用いて、引数 1 で指定したフィールドの内容を引数 2 の値と比較します。

Eval 比較演算子が指定された場合は、引数 1 フィールドは無視されます。代わりに、引数 2 の値を評価式と見なし、直接評価を実行します。

- 引数 2 - 引数 1 で指定されたフィールドの内容と比較する値を入力します。たとえば、**result** フィールドの値が 5 より大きいときに成功と見なす場合、引数 1 に文字列「**result**」を、オペレータフィールドに「>」を、引数 2 フィールドに値「5」を入力します。

オペレータフィールドで **Eval** が指定された場合は、このフィールドには評価式が含まれている必要があります。評価式には **Visual Basic** の式を使用し、結果が **true** または **false** のいずれかで評価されます。たとえば、以下の式を指定 (すべて引数 2 に入力します) することで、上記と同様に **result** フィールドを値 5 と「**and**」を使用して比較することができます。

```
eventDetails("result") > 5
```

この例で評価されているのは、**result** フィールドの内容と値 5 との比較です。もちろん、同様の方法でより複雑な式を評価することもできます。以下により複雑な式の例を示します。

```
Tools.NetTime(eventDetails("startDate"),eventDetails("endDate")) >
eventDetails("MaxTimeSpan")
```

この例の式では、2 つのフィールドの値に基づく **Tools** オブジェクトの **NetTime** メソッドの結果を 3 番目のフィールドの値に照らして比較しています。

- 行の論理関係 - 現在の行と次の行との比較における論理関係を入力します。複数の比較を含めて実行することができます。比較または評価に使用する式は表の個別の行に記入します。このフィールドにより、比較の結果が互いにどのように関係があるかが定義されます。各比較間の関係は通常の論理的な優先度ルールには従いません。常に表に記述されている順序に従って評価されます。このフィールドでの有効な値は「and」または「or」のみです。表に含まれる行が 1 のみの場合には、このフィールドは無視されるので、空のまま構いません。

Success Calculation モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
IgnoreTimeslots	(オプション、文字列 [True/False]) - タイムスロット期間外に発生するイベントを処理するかどうかを指定します。
Debug	(オプション、文字列 [True/False]) - ログにデバッグ コメントを出力するかどうかを指定します。
AggregationType	(オプション、文字列 [SUCCESSCOUNT,PERCENTOFSUCCESS]) - 成功率を算出するか成功したエントリの数を算出するかを指定します。
SuccessCriteria	(必須、テーブル) - イベントの成功または失敗を判定する方法を定義します。
FilteringCriteria	(オプション、テーブル) - イベントを処理の対象とするか対象外とするかを判定する方法を定義します。
EventIdFieldName	(オプション、文字列) - 繰り返しイベントを認識するためのイベント ID として使用するフィールドの名前です。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Parameters Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Aggregation Library (AL)	一般的な計算の実行に使用されるライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Parameters Overlay - Basic Actions Library (POBAL)	特定のパラメータローディング関数が含まれるライブラリで、実際の情報のロード用に Parameters Kernel Library をコールします。
Distinct Entity Library (DEL)	指定したエンティティがすでに参照済みかどうかをコール元に通知するためのライブラリです。
Filtering Criteria Evaluation Library (FCEL)	ユーザの設定に従ってイベントのフィルタリングを行うためのライブラリです。
Success Criteria Evaluation Library (SCEL)	現在のイベントを成功とみなすか失敗とみなすかを判定するためのライブラリです。

Incident Management Business **ビジネス ロジック モジュール**

以下のセクションでは、ヘルプデスク サービスに共通のメトリックで使用される汎用のビジネス ロジック モジュールについて説明します。これらのルールはインシデント管理、問題管理および変更管理のメトリックでの計算に適用されます。

インシデント管理モジュールの計算は、一定のインシデントライフ サイクルに基づいて行われます。インシデントのライフ サイクルは、1 つのインシデントが受け取るさまざまなステータスに反映されます。

以下の表にインシデントステータス、各インシデントステータスの意味、および計算結果への影響の一覧を示します。インシデントライフ サイクルは問題と変更の両方に適用されます。

チケット ステータス	説明
Open	インシデントのライフサイクル中で 1 回生成されます。作成されるのはチケットがヘルプデスク システム最初に登録されたときです。

Reopen	Closed ステータスになった後にのみ 1 回生成できます。再オープンイベントは計算では新規インシデントと見なされます。
InProcess	インシデントのライフ サイクルで 1 回生成されます。チケットの作業が開始された時に生成され、実際に誰かが解決のために調査および作業を行っている状態です。
Pending	顧客からの返答や情報を待つ、またはサプライヤやその他依存しているサードパーティから機器を待つといったような、合計解決期間には含まれないと見なされるような何らかのアクティビティ状況が開始したときに生成されます。1 つのインシデントで複数回生成することが可能です。インシデントの保留状態の時間を終了するには、再度 InProcess イベントを受け取る必要があります。
Resolved	インシデントのライフ サイクルで 1 回生成されます。インシデントの解決策が提供され、顧客に伝えられた時に生成されます。
Closed	インシデントのライフ サイクルで 1 回生成されます。

もしインシデントで優先度が変更されている場合は、Close イベントが変更日にトリガされ、新しい優先度を持つ Open イベントがトリガされる必要があります。

Incident Management イベントタイプ

以下の Incident Management イベントタイプは、管理対象の各インシデントのインシデントライフ サイクル中でのインシデントのステータスを表しています。

- Incident - Open
- Incident - ReOpen
- Incident - In process
- Incident - Pending
- Incident - Resolved
- Incident - Closed

Change Management イベントタイプ

Change Management イベントタイプは、管理対象の各変更のライフ サイクル中での変更リクエストのステータスを表しています。

- Change - Open
- Change - ReOpen
- Change - In process
- Change - Pending
- Change - Resolved
- Change - Closed

Problem Management イベントタイプ

以下の Problem Management イベントタイプは、管理対象の各問題のライフ サイクル中での問題のステータスを表しています。

- Problem - Open
- Problem - ReOpen
- Problem - In process
- Problem - Pending
- Problem - Resolved
- Problem - Closed

Incident/Change/Problem イベント タイプの構成

以下の表に示すように、インシデント イベントの構成はすべて同一です。

#	名前	フィールド タイプ	フィールドの説明	計算に必要
1	ID	文字列	インシデントの ID	はい
2	Priority	文字列	使用できる優先度レベルは 3 つあります。	いいえ
3	Customer	文字列	顧客 ID	いいえ
4	Service	文字列	サービス ID	いいえ
5	LocationName	文字列	複数サイト環境での場所の名前	いいえ

以下のサービスドメインがこのイベント構成を利用します。

- Incident Management
- Problem Management
- Change Management

Percent Of Reopened Tickets Module モジュール

このモジュールでは、計算期間内に再オープンされたチケット数を計算します。パーセンテージは Priority X ReOpen チケット イベント数をカウントすることによって計算されます。

モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
WhatToCalculate	期間の最後の最終出力を決定します。
Debug	デバッグ モードであるかどうかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。

Entity Counter Library (ECL)	エンティティカウンタの追跡に使用するライブラリ。
Parameter Kernel Library (PKL)	モジュールパラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Quick Metric Filtering (QM)	クイックメトリック機能で必要となるフィルタリングモジュール。
Quick Metric Mapping (QM)	クイックメトリック機能で必要となるマッピングモジュール。

モジュール計算 (WhatToCalculate パラメータ)

- **Percent** - 計算期間中にクローズされたすべてのインシデントで再オープンされたチケットのパーセンテージを計算します。
- **Count** - 計算期間中にクローズされたすべてのインシデントで再オープンされたチケットの数を計算します。

イベント登録

- イベントタイプの一覧については「[Incident Management ビジネス ロジック モジュール \(P. 54\)](#)」を参照してください。

Resolution Time モジュール

インシデント/問題/変更の解決時間に基づいて結果を計算するメトリックはすべて、このモジュールを使用し、同一の計算メソッドを使用します。

インシデントの解決時間は、インシデントがオープンされた時点から最終解決策が提供されるまでの経過時間として算出されます。インシデントが保留中のステータスであった時間は解決時間から除外されます。また、タイムスロット期間外の時間も解決時間から除外されます。

計算での前提

- インシデントはクローズ イベントを 1 回受け取ります。クローズ イベント以降のイベントはいずれも別のインシデントまたは新規のインシデントと見なされます。
- インシデントがオープン イベントを受け取らなかったような場合 (通常は契約の発効日前に開始されたなどの理由のため)、解決時間のカウントは契約の発効時から開始します。
- インシデントの最初のイベントがオープン イベントでないと、タスクは停止状態とマークされます。
- Closed イベントは 1 件のインシデントで受信できる最後のイベントです。
- タイムスロット期間外の時間、または例外時間は解決時間にはカウントされません。
- インシデントに複数のオープン イベントがある場合、最初のものが開始時間である見なされます。
- 閉じられたチケットがない計算期間がある場合、その期間の結果は NULL になります。
- インシデントが計算時に考慮されるのは、そのインシデントがクローズされた期間であって、解決された期間ではありません。

モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
TimeUnit	Threshold の時間単位です。たとえば、しきい値が 3 日であるとき、TimeUnit は「日」です。

TargetTime	各インシデントの比較を行う範囲を示す解決時間のしきい値です。このしきい値内で解決されたインシデントのパーセンテージが計算されます。たとえば、しきい値が3日であるとき、TargetTimeは「日」です。
WhatToCalculate	期間の最後の最終出力を決定します。たとえば、しきい値が3日であるとき、TargetTimeは「日」です。
Debug	デバッグ モードであるかどうかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Time Counter Library (TCL)	エンティティ時間カウンタの追跡に使用するライブラリです。
Aggregation Library (AL)	一般的な計算の実行に使用されるライブラリ。
Parameter Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

- Avg - 計算期間中にクローズされたすべてのインシデントの平均解決時間を計算します。
- %WithinThreshold/Percent - 計算期間内にクローズされたインシデントに関して、解決時間がしきい値内であるインシデントのパーセンテージを計算します。
- CNTWithinThreshold - 計算期間内にクローズされたインシデントに関して、解決時間がしきい値内であるインシデントの数を計算します。
- Min - 計算期間中で最短の解決時間でクローズされたインシデントを決定します。
- Max - 計算期間中で最長の解決時間でクローズされたインシデントを決定します。

イベント登録

- イベントタイプの一覧については「[Incident Management ビジネス ロジック モジュール \(P. 54\)](#)」を参照してください。

Response Time モジュール

インシデントの応答時間に基づいて結果を算出するメトリックは、各インシデントの応答時間の計算にすべて同一の計算方法を使用します。

応答時間は、インシデントがオープンされてから、それに対して処理が行われるまでの経過時間として計算されます。インシデントが保留中のステータスであった時間は応答時間から除外されます。また、タイムスロット期間外の時間も応答時間から除外されます。

モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
TimeUnit	Threshold の時間単位です。たとえば、しきい値が 3 日であるとき、TimeUnit は「日」です。
TargetTime	各インシデントの比較を行う範囲を示す解決時間のしきい値です。このしきい値内で解決されたインシデントのパーセンテージが計算されます。たとえば、しきい値が 3 日であるとき、TimeUnit は「日」です。

WhatToCalculate	期間の最後の最終出力を決定します。たとえば、しきい値が 3 日であるとき、TargetTime は「日」です。
Debug	デバッグ モードであるかどうかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Time Counter Library (TCL)	エンティティ時間カウンタの追跡に使用するライブラリです。
Aggregation Library (AL)	一般的な計算の実行に使用されるライブラリ。
Parameter Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

- Avg - 計算期間中にクローズされたすべてのインシデントの平均応答時間を計算します。
- %WithinThreshold - 計算期間中にクローズされたインシデントのうち、応答時間がしきい値内にあるインシデントの割合を算出します。
- Min - 計算期間中に最短応答時間でクローズされたインシデントを特定します。
- Max - 計算期間中に最長応答時間でクローズされたインシデントを特定します。

イベント登録

- イベントタイプの一覧については「[Incident Management ビジネス ロジック モジュール \(P. 54\)](#)」を参照してください。

Timeslot Events Sender モジュール

Resolution Time ビジネス ロジック モジュールは、イベントの生成元である顧客のタイムスロットを考慮しないため、Timeslot Events Sender ビジネス ロジック モジュールおよび **Resolution By Timeslots Extension Library** が作成されました。ユーザの会社が多国籍で、処理しているチケットを顧客のタイムフレームでの一定の日数内に解決する必要がある場合、チケットの生成元である顧客の場所を考慮する必要があります。解決時間では、特定の場所のタイムゾーンおよびタイムスロットが考慮される必要があります。その場所は実際の計算を実行しているメトリックがある場所とは異なる場合があります。

このモジュールは、再利用イベントを使用してタイムスロットの開始イベントおよび終了イベントを送信します。このモジュールを実装するには、それを含むメトリックを作成する必要があります。送信する各メトリックは、場所に固有である必要があります。場所の名前がメトリック名の一部に含まれる必要があります。メトリックには場所に関連するタイムスロットおよびタイムゾーンが必要です。また、**LocationId** という名前の単一のパラメータに値が入力されている必要があります。このパラメータの内容は場所の名前であり、この場所に関連したイベントのイベント詳細に表示されます。

モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
LocationId	メトリック用の場所の一意の名前。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Parameters Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。

Total Number of Incidents モジュール

計算期間内に受信されたイベントの数をカウントします。カウントを行う必要があるどのタイプのイベントでもこのモジュールで使用することができます。たとえば、計算期間中にオープンされたインシデント数をカウントする必要がある場合は、**Open** イベントに対して登録を行います。期間中にクローズされたインシデントの数をカウントする必要がある場合は、登録は **Closed** のイベントに対して行います。計算式は、イベントの ID という名前のフィールドのみを使用します。そのため、この値フィールドを持つ任意のイベントがカウントに使用できます。

計算での前提

- 計算期間中に受け取ったすべてのインシデントがカウントされます。
- タイムスロット期間外に受け取ったインシデントも同様にカウントされます。
- インシデントは、インシデントの ID 基に 1 回数えられます。同一のイベントタイプから受け取った複数のイベントのインシデント ID が 1 つである場合は、単一のものとしてカウントされます。

モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
WhatToCalculate	期間の最後の最終出力を決定します。
Debug	デバッグ モードであるかどうかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Time Counter Library (TCL)	エンティティ時間カウンタの追跡に使用するライブラリです。
Aggregation Library (AL)	一般的な計算の実行に使用されるライブラリ。
Parameter Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	一部の一般的なパラメータの検証および入力関連のライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。

Quick Metric Filtering (QM) クイック メトリック機能で必要となるフィルタリング モジュール。

Quick Metric Mapping (QM) クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

COUNT - 選択したイベントタイプに応じて簡易的なエントリのカウントを行います。

イベント登録

イベントタイプの登録はカウント対象のイベントに合わせて行います。このときイベントタイプは ID の値フィールドを持っている必要があります。

たとえば、任意の優先度でオープンしているイベントをカウントするには、登録を Incident - Open イベントタイプに対して行います。

Within X Business Days モジュール

このモジュールには、イベントを処理し、結果を計算し、期間の管理を行うのに必要な機能が含まれています。期間の管理は、指定された期間内のすべてのチケットを解決するのに要する日数を計算するために必要です。「X 日以内」の計算と「同一日/翌日」の両方の機能を提供します。

各チケットの解決時間は、チケットがオープンされた時点から最終解決策が提供されるまでの経過時間として算出されます。チケットが保留ステータスであった時間はカウントされません。また、タイムスロット期間外の時間もカウントされません。

モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
WhatToCalculate	計算の方法を決定します。
Debug	デバッグ モードであるかどうかを決定します。
DaysThreshold	これは解決しきい値(日数)です。
CountWholeDays	オープン時間を考慮に入れて計算するか、1 日の中のタイムスロット期間内の一瞬であっても 1 日としてカウントするかを決定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Business Day Counting Library (BDCL)	チケット ライフサイクル内の営業日をカウントするライブラリ。
Parameters Overlay - Basic Action Library (POBAL)	ログ パラメータのような複数のライブラリによって使用される一般的なパラメータを含むライブラリ。
Parameters Overlay - Business Days Library (POBDL)	営業日数のカウント機能に固有なパラメータを含むライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

イベント登録

- イベントタイプの一覧については「[Incident Management ビジネス ロジック モジュール \(P. 54\)](#)」を参照してください。

Service Desk Management **ビジネス ロジック モジュール**

以下のセクションでは、コール センター サービスに共通のメトリックで使用されるビジネス ロジック モジュールについて説明します。

Service Desk 管理のイベントタイプ

Call Center service desk management イベントタイプでは、特定のセンターまたはオペレータのコール情報を提供します。

Call Center イベントタイプの構成

以下の表に Call イベントタイプの構成を説明します。

#	名前	説明
1	EventTimestamp	タイムスタンプ。

#	名前	説明
2	CallCenter	コール センター/オペレータの名前(場所や別のプロバイダである場合もあります)。
3	TotalCalls	指定のタイム スタンプの時刻にコールセンターで受けたコール数。
4	Answered	対応したオペレータ(エージェント)に接続されたコール総数。
5	Abandoned	対応されなかったコール総数。サービスが提供される前に破棄されたコール総数。
6	AnsweredWithinX Seconds_Count1	対応するバケットで示される制限時間内に対応されたコール数を表す数値です(制限 1)。
7	AnsweredWithinX Seconds_Limit1	秒数を表す数値で、その秒数内にコールに対応します(制限 1)。
8	AnsweredWithinX Seconds_Count2	対応するバケットで示される制限時間内に対応されたコール数を表す数値です(制限 2)。
9	AnsweredWithinX Seconds_Limit2	秒数を表す数値で、その秒数内にコールに対応します(制限 2)。
10	AnsweredWithinX Seconds_Count3	対応するバケットで示される制限時間内に対応されたコール数を表す数値です(制限 3)。
11	AnsweredWithinX Seconds_Limit3	秒数を表す数値で、その秒数内にコールに対応します(制限 3)。
12	CallsAfter60s	エージェントによって対応されたコール数。ルーティング メカニズムでコールが処理されてからエージェントによって対応されるまでにかかった時間(待ち時間と呼び出し時間の合計)と関連します。
13	AbandBefore10s	サービスが提供される前に破棄されたコールの総数です。
14	AbandBefore30s	サービスが提供される前に破棄されたコールの総数です。
15	AbandBefore60s	サービスが提供される前に破棄されたコールの総数です。
16	AbandAfter60s	サービスが提供される前に破棄されたコールの総数です。
17	AvgDurCallProc	対応したコールの処理時間の平均です。電話の処理手続きは、エージェントがコールに対応してから会話を終了するまでの間で計算されます。
18	AvgWaitAnsCalls	リソースに接続したコールが順番待ちをしていた時間の平均です。破棄されたコールおよび待機タイマーのオーバー フローのためにリダイレクトされたコールは考慮されません。

#	名前	説明
19	EscalatedCalls	インシデント管理システムでインシデントがオープンされたコールのパーセンテージです。

Call Center モジュール

イベントソースから受け取ったデータは、フィールド名で定義されたサマリの形で集計済みです。このモジュールを使用して、要求された特定のサマリ(たとえば破棄されたコール数)をコールの総数で割ることで、サマリの内容をパーセンテージに変換することができます。

コールのパーセンテージを計算するすべての計算式は同様の計算を実行します。ただし、イベントから適切なフィールドを選択することで、実行する計算に適したサマリが使用されます。

タイムスロット期間外のタイムスタンプを持つイベントは無視されます。

「X 秒以内で対応されたコール数」機能での計算では、モジュールは、バケットのサイズ(秒数)が含まれるフィールド、および結果(数のサマリ)を含むフィールドのペア 3 つに基づく動的なシステムを使用します。この方法では、秒数はフィールド名の一部ではなく、任意の秒数を処理することができます。この動的なシステムを使用するには、2 つのパラメータが使用されます。1 つはこの機能をオンにするもので、もう 1 つは使用するバケットをメトリックに知らせるためのものです。その後、メトリックは指定されたサイズのバケットを探し、対応するフィールドのコール数を使用します(たとえば、バケットが

`AnsweredWithinXSeconds_Limit2` フィールドで見つかった場合、`AnsweredWithinXSeconds_Count2` フィールドの値が使用されます)。この機能では、指定されたバケットサイズのフィールドが 1 つのみ常に存在していることを想定しています。それが見つからない場合はエラーが発生します。

イベントのバケットはすべてまたは一部のみが使用されている場合があります。各メトリックは異なるバケットサイズである必要があり、このサイズはイベント内に存在している必要があります。つまり、各イベントには同時に最大 3 つまでの異なるバケットサイズを格納しておけることになります。

モジュール インターフェース

パラメータを以下の表に示します。

名前	説明
CallType	コール累積数用のイベントタイプから関連フィールドを選択するために計算式が使用します。利用可能なオプションは次のとおりです： ANSWERED、ABANDONED、ESCALATED、OVERALL、CALLSAFTER60S、 ABANDBEFORE10S、ABANDBEFORE30S、ABANDBEFORE60S、 ABANDAFTER60S、PERCENTESCCALLS

WhatToCalculate	計算にどの計算式を使用するかを決定します。利用可能なオプションは次のとおりです: Percent、Count、WaitingTime、ProcessingTime。詳細については下記を参照してください。
Debug	デバッグ モードであるか、またデバッグ メッセージをログに送信するかを決定します。利用可能なオプションは次のとおりです: True または False
RespondBucketLimit	バケットリミット用のイベントタイプから関連フィールドを選択するために計算式が使用します。値は数値で、秒数を指定します。
AnsweredWithinTarget	AnsweredWithin ロジックを使用するかどうかをシステムに通知します。True に設定されている場合、CallType パラメータは無視されます。利用可能なオプションは次のとおりです: True または False
TimeUnit	計算に使用する時間単位をシステムに通知します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Parameter Overlay - Call Center Library (POCCL)	コール センター機能に固有なパラメータを含むライブラリ。
Parameter Overlay - Basic Action Library (POBAL)	ログ パラメータのような複数のライブラリによって使用される一般的なパラメータを含むライブラリ。
Callback Kernel Library (CKL)	コールバック機能を有効にするライブラリ。
Quick Metric Filtering (QM)	クイック メトリック機能で必要となるフィルタリング モジュール。
Quick Metric Mapping (QM)	クイック メトリック機能で必要となるマッピング モジュール。

モジュール計算 (WhatToCalculate パラメータ)

- **Percent** - 計算期間での総コール数の内、特定タイプからのコールのパーセンテージ。たとえば、廃棄されたコールのパーセンテージは、 $\frac{\sum(\text{廃棄されたコールの数})}{\sum(\text{コールの数})} * 100$ として計算されます。
- **Count** - 計算期間中のコールの総数を計算します。
- **WaitingTime** - 平均待ち時間を計算します。計算式は $\frac{\sum(\text{イベント待ち時間})}{\sum(\text{イベントの数})}$ になります。
- **ProcessingTime** - 平均処理時間を計算します。計算式は $\frac{\sum(\text{イベント処理時間})}{\sum(\text{イベントの数})}$ になります。

イベント登録

Call Center モジュールを使用するメトリックはすべて Call Center イベントタイプに登録されます。

Library モジュール

事前定義済みコンテンツ パッケージにはライブラリ モジュールも含まれ、完全なビジネスロジック モジュールを使用するためのユーティリティセットが提供されます。

ライブラリの一覧を以下に示します。

ライブラリ名	説明
Aggregation Library	一般的な計算の実行に使用されるライブラリ。
Business Day Counting Library	実際のタイムスタンプ操作を行うライブラリで、営業日の開始および終了の時間を算出します。詳細については、「 Business Day Counting Library (P. 74) 」を参照してください。
Callback kernel Library	コールバック機能を有効にするライブラリで、関数が実装されていればそれが呼び出されます。実装されていない場合は、コールは行われずエラーは発生しません。
Criteria Evaluation Kernel Library	フィルタリング ライブラリまたは成功ライブラリのいずれかに関して、実際の基準評価を行うライブラリです。
Distinct Entity Library	指定したエンティティがすでに参照済みかどうかをコール元に通知するためのライブラリです。
Entity Counter Library	エンティティカウンタの追跡に使用するライブラリ。

Filtering Criteria Evaluation Library	ユーザの設定に従ってイベントのフィルタリングを行うためのライブラリです。イベントハンドラは各イベントについてこの機能呼び出し、イベントを処理するかどうかを決定します。
Log Library	拡張ログ機能のライブラリ。
Parameters Kernel Library	パラメータの読み取り処理の中核部を含むライブラリです。内部の関数には、デフォルト値、許可値、パラメータがテーブルであるかどうかといった、パラメータのロードに必要なすべての情報が与えられます。メトリックパラメータ情報に安定してアクセスすることが可能です。
Parameter Library	モジュールパラメータの検証および入力関連のライブラリ。
Parameters Overlay - Basic Action Library	特定のパラメータローディング関数が含まれるライブラリで、実際の情報のロード用に Parameters Kernel Library をコールします。
Resolution By Timeslots Extension Library	場所に固有なタイムスロット機能を解決モジュールに追加します。詳細については、「 Resolution By Timeslots Extension Library 」を参照してください。
Success Criteria Evaluation Library	現在のイベントを成功とみなすか失敗とみなすかを判定するためのライブラリです。 Criteria Evaluation Library をコールします。イベントハンドラは、フィルタリングを通過したすべてのイベントについてこの関数を呼び出し、イベントを成功と見なすかどうかを決定します。
Time Counter Library	エンティティ時間カウンタの追跡に使用するライブラリです。
Within X Business Days By Timeslots Extension Library	複数のソースタイムスロットを処理できるように Within X Business Days モジュールを拡張するライブラリです。詳細については、「 Within X Business Days By Timeslots Extension Library (P. 74) 」を参照してください。

Resolution By Timeslots Extension Library

このモジュールは、場所に固有なタイムスロット機能を解決モジュールに追加します。この機能を有効にするには、メトリックの解決モジュールの他に拡張ライブラリを含める必要があります。拡張ライブラリには新しいパラメータが 2 つあり、それらの値が入力されている必要があります。

- **RegistrationList** - テーブルパラメータです。テーブルには **LocationId** という名前の列が 1 つあります。このパラメータには、メトリックに関連した場所でのタイムスロットの開始および終了イベント(**Timeslot Events Sender** モジュールによって作成)を送信するすべてのメトリック名のリストが含まれている必要があります。リストには、場所の名前ではなくイベントを送信するメトリックの名前が含まれる必要があります。たとえば、メトリックの名前のプレフィックスが「**Location -**」であるなら、リストには「**Location - France**」および「**Location - Spain**」のような値が含まれている必要があります。このリストは、指定した場所に関するタイムスロットの開始および終了イベントの登録および受信に使用されます。
- **EventLocationId** - このパラメータは、処理対象の各イベントの詳細で、どのように場所情報を検索するかをシステムに通知します。これには、以下の 3 つの方法のいずれかが使用されます。
 - 場所情報が含まれるイベントタイプフィールドの名前。この場合、パラメータのコンテンツは対応するフィールドの名前です。
 - イベント詳細に含まれていないイベント固有の情報。この情報にアクセスするには、以下のように最初の文字が **@** である必要があります。
 - **@resource** – イベントを送信したリソースの名前。
 - **@eventtype** – イベントを送信したイベントタイプの名前。
 - **@sendermetric** – イベントを送信したメトリックの名前。
 - **VBS** で評価可能な任意の評価式を指定することができます。これは、複数のフィールドにアクセスしたり、送信リソースのカスタム属性情報にアクセスしたりするために使用できます。この機能を使用するには、以下のように最初の文字が「**=**」(等号)である必要があります。たとえば、イベントを送信したリソースのカスタム属性を抽出するには、以下のように記述します。
`=eventDetails.CustomAttribute("MyAttribute")`
これは、イベントを送信したリソースの **myAttribute** カスタム属性の値をイベントの場所の ID として使用します。

Within X Business Days By Timeslots Extension Library

このライブラリは複数のソースタイムスロットを処理できるように **Within X Business Days** モジュールを拡張します。この機能を有効にするには、メトリックの解決モジュールの他に拡張ライブラリを含める必要があります。拡張ライブラリには新しいパラメータが 2 つあり、それらの値が入力されている必要があります。

- **RegistrationList** - テーブル パラメータです。テーブルには **LocationId** という名前の列が 1 つあります。このパラメータには、メトリックに関連した場所でのタイムスロットの開始および終了イベント(**Timeslot Events Sender** モジュールによって作成)を送信するすべてのメトリック名のリストが含まれている必要があります。リストには、場所の名前ではなくイベントを送信するメトリックの名前が含まれる必要があります。たとえば、メトリックの名前のプレフィックスが「**Location -**」であるなら、リストには「**Location - France**」および「**Location - Spain**」のような値が含まれている必要があります。このリストは、指定した場所に関するタイムスロットの開始および終了イベントの登録および受信に使用されます。
- **EventLocationId** - このパラメータは、処理対象の各イベントの詳細で、どのように場所情報を検索するかをシステムに通知します。これには、以下の 3 つの方法のいずれかが使用されます。
 - 場所情報が含まれるイベントタイプフィールドの名前。この場合、パラメータのコンテンツは対応するフィールドの名前です。
 - イベント詳細に含まれていないイベント固有の情報。この情報にアクセスするには、以下のように最初の文字が **@** である必要があります。
 - **@resource** – イベントを送信したリソースの名前。
 - **@eventtype** – イベントを送信したイベントタイプの名前。
 - **@sendermetric** – イベントを送信したメトリックの名前。
 - **VBS** で評価可能な任意の評価式を指定することができます。これは、複数のフィールドにアクセスしたり、送信リソースのカスタム属性情報にアクセスしたりするために使用できます。この機能を使用するには、以下のように最初の文字が「**=**」(等号)である必要があります。たとえば、イベントを送信したリソースのカスタム属性を抽出するには、以下のように記述します。
`=eventDetails.CustomAttribute("MyAttribute")`
これは、イベントを送信したリソースの **myAttribute** カスタム属性の値をイベントの場所の ID として使用します。

Business Day Counting Library

このモジュールは実際のタイムスタンプの計算を行い、営業日の開始および終了の時間を算出します。2つの異なるロジックが提供されます。

- **X日以内** - これは、日数をカウントする際にチケットのオープン時間を考慮に入れます。たとえば、チケットが **14:00** にオープンされ、翌営業日の **13:00** にクローズされた場合、これを **1日** とカウントされます。チケットが翌営業日の **15:00** にクローズされた場合は、**2日** とカウントされます。
- **同一日、翌日** - これはチケットオープン時間を考慮しません。アクティブなタイムスロットが存在している日(つまり営業日)であれば、該当する時間がわずかであってもその日を **1日** としてカウントします。たとえば、チケットが日付 **1** のタイムスロット内の最後の **1秒** でオープンされた場合でも、その日は **1日** とカウントされます。チケットが日付 **1** のタイムスロット後にオープンされた場合は、その日はカウントされません。「同一日」とマークされているチケットの場合、チケットはその日の終わりまでにクローズする必要があります。チケットのオープン直後の時間が含まれるタイムスロットによって、どちらの日になるかが決定されることに注意してください。今日のタイムスロット終了後にオープンされたチケットは翌日からアクティブであると考えられます。従って今日のタイムスロット終了後にオープンされた場合、「同一日」のチケットは翌日の終業時までにはクローズする必要があります。オープン日の深夜 **0時** 以降にオープンされたチケットは **2日** とカウントされます。

どちらの営業日ロジックも、同一のモジュールに基づいており、違っている点はパラメータの設定のみです。

- **単一タイムスロット** - 計算するメトリック自体のタイムスロット/タイムゾーンが単一であることを想定しています。以下のパラメータを実装する必要があります。
 - **WhatToCalculate** - このパラメータを使用して、しきい値の範囲内でのチケット数を取得するか、チケットの合計をもとにしきい値の範囲内でのチケットのパーセンテージを取得するかを決定できます (PERCENT/CNTWITHINTHRESHOLD)。
 - **CountWholeDays** - このパラメータは、ユーザが「同一日/翌日」機能を使用するか(TRUE)、「X営業日以内」機能を使用するか(FALSE)を決定します。
 - **DaysThreshold** - これは、どれだけの日数で成功と見なすかをシステムに通知します。「同一日」機能では、**1日** 目も日数にカウントされます。従って、**Same day = 1**、**Next day = 2** となります。「X日以内」については、日数はより直観的に理解できます。したがって、「**3日以内**」の場合、値は **3** である必要があります。

注: 「同一日/翌日」機能は、実際にはこれら 2 つの値に制限される訳ではなく、たとえば 3 日目の終わりまでのように実装することもできます。

- 複数タイムスロット - 上記の単一タイムスロットに加え、新しい機能では複数タイムスロットのイベントがサポートされます。このサポートは、**Within X Business Days By Timeslots Extension Library** という名前の追加の拡張ライブラリによるものです。このライブラリはメトリック自体に接続されます。これによってタイムスロット機能が追加され、さらに追加のパラメータ ペアが必要になります。
 - **RegistrationList - LocationId** という名前の単一のフィールドを持つテーブル パラメータです。このテーブルには、同一契約内に配置される必要がある場所のタイムスロット イベントを送信するメトリックのリストが含まれます。
 - **EventLocationId** - 受信イベントの場所を決定するためにシステムが実行する必要のあるものの定義です。

また、この機能を有効にするために、新しいイベントタイプが追加され、チケットライフサイクル間での夏時間の変更を処理できるようになりました。

このイベントタイプは **LocationDstChangeNotification** という名前で、以下の仕様に定められています。

- **LocationId** - 文字列で指定します。
- **DistanceFromUtc** - 整数で指定します。

モジュール依存関係

含まれるモジュールを以下の表に示します。

名前	説明
Log Library (LL)	拡張ログ機能のライブラリ。
Parameters Kernel Library (PKL)	モジュール パラメータの検証および入力関連のライブラリ。

第 6 章：概要

このドキュメントには、事前に定義したコンテンツのリリースノートが含まれます。以下の章ではリリースについて詳細を説明し、メインのドキュメントを補足するその他の情報を提供します。

このリリースでは品質、信頼性、パフォーマンスが改善され、新しい機能が追加されました。

第 7 章：インストールとアップグレード

詳細なインストール手順については、「CA Business Service Insight 事前定義済みコンテンツ ガイド」を参照してください。

第 8 章：その他の拡張機能

このリリースの事前定義済みコンテンツに新しい拡張機能は追加されていません。

第 9 章: 既知の問題

このリリースの事前定義済みコンテンツで問題は見つかりませんでした。

第 10 章：概要

このドキュメントには、事前に定義したコンテンツのリリースノートが含まれます。以下の章ではリリースについて詳細を説明し、メインのドキュメントを補足するその他の情報を提供します。

このリリースでは品質、信頼性、パフォーマンスが改善され、新しい機能が追加されました。

第 11 章: インストールとアップグレード

詳細なインストール手順については、「CA Business Service Insight 事前定義済みコンテンツガイド」を参照してください。

第 12 章：その他の拡張機能

このリリースの事前定義済みコンテンツに新しい拡張機能は追加されていません。

第 13 章：既知の問題

このリリースの事前定義済みコンテンツで問題は見つかりませんでした。

索引

A

- Availability and Failure Statistics モジュール - 40
- Availability Management 関数リスト - 31
- Availability Management ビジネス ロジック モジュール - 38
- Availability Management メトリック リスト - 18
- Average Measurements モジュール - 44

B

- Business Day Counting Library - 75

C

- CA Business Service Insight および ITILv3 事前定義済みパッケージ - 12
- Call Center モジュール - 69
- Capacity & Performance Management メトリック リスト - 19
- CA への連絡先 - 3
- Change Management メトリック リスト - 22
- Configuration Management メトリック リスト - 23
- Count Unique モジュール - 45

E

- Elapsed Time モジュール - 46

G

- Generic Calculation モジュール - 47
- Generic Formulas – Advanced 関数リスト - 32
- Generic Formulas – Elapsed Time 関数リスト - 33
- Generic Samples テンプレート フォルダ - 17
- Generic Samples メトリック リスト - 17

I

- Incident Management – Multi Timeslot メトリック リスト - 25

- Incident Management Business ビジネス ロジック モジュール - 54

- Incident Management 関数リスト - 32
- Incident Management メトリック リスト - 24
- ITILv3 コンテンツ参照 - 15
- ITILv3 実装ノート - 29
- ITIL プロセス モデリング ガイドライン - 11

L

- Library モジュール - 71

P

- Percent Of Reopened Tickets Module モジュール - 57
- Problem Management メトリック リスト - 26

R

- Resolution By Timeslots Extension Library - 73
- Resolution Time モジュール - 59
- Response Time モジュール - 61

S

- Service Delivery テンプレート フォルダ - 18
- Service Desk Management ビジネス ロジック モジュール - 66
- Service Desk Management メトリック リスト - 27
- Service Support テンプレート フォルダ - 21
- Success Calculations モジュール - 49

T

- Timeslot Events Sender モジュール - 63
- Total Number of Incidents モジュール - 64

W

- Within X Business Days By Timeslots Extension Library - 74
- Within X Business Days モジュール - 65

あ

インストールとアップグレード - 79, 87

か

関数パッケージ - 13

関数パッケージコンテンツ参照 - 31

基準パラメータの構成 - 51

既知の問題 - 83, 91

さ

全般機能 - 37, 50

その他の拡張機能 - 81, 89

た

対象読者 - 10

は

汎用ビジネスロジックモジュール - 42

ビジネスロジックモジュール - 35

本ガイドの使用方法 - 10

漢字

概要 - 9, 77, 85