



**Interaction Concentrator 7.6**

# **User's Guide**

このマニュアルに記載されている内容は所有権付きの機密情報であり、Genesys Telecommunications Laboratories, Inc. の書面による事前の承諾がない限り開示も複製もできません。

Copyright © 2006–2008 Genesys Telecommunications Laboratories, Inc. All rights reserved.

## Genesys について

Genesys Telecommunications Laboratories, Inc. (Alcatel-Lucent の子会社) は、コールセンター用のソフトウェアを専門に扱っています。インタラクション (情報のやり取り) を適切に進めることが事業の成功をもたらす、企業の評価につながると Genesys は認識しています。この認識に基づいて Genesys が提供する顧客サービス ソリューションは、80 か国に及ぶグローバル 2000 企業、政府機関、電気通信サービス事業者を対象に、毎日 1 億件を超える顧客のインタラクションを管理しています。音声、電子メール、および Web チャネルを使用した洗練されたルーティングおよびレポートにより、お客様は初めて、利用可能な最善のリソースに迅速に接続できるようになりました。Genesys は、顧客サービス、ヘルプデスク、オーダーデスク、回収業務、電話によるセールスやサービス、ワークフォース管理などの業務向けにソリューションを提供しています。詳細については、<http://www.genesyslab.com> にアクセスしてください。

どの製品にも個別のオンライン マニュアルが用意されていて、Genesys テクニカル サポートの Web サイト上か、または Genesys に連絡して入手できる Genesys Documentation Library DVD で閲覧できます。詳細については、営業担当者にお問い合わせください。

## 注意

このマニュアルでは、発刊時の完全かつ正確な内容記載を実現するよう細心の注意が払われていますが、もしなんらかの誤りがあっても Genesys Telecommunications Laboratories, Inc. はその責任を負うことはできません。このマニュアルに記載されている情報の変更または修正 (あるいはその両方) は、今後のバージョンに反映されます。

## システムのセキュリティに関するお客様の責任

ご使用のシステムのセキュリティについては、お客様が責任を負うものとします。無断使用を防ぐための製品管理についても、お客様が責任を負うものとします。システム管理者は、この製品に用意されているマニュアルをすべて読んで使用可能な機能を完全に把握しておいてください。そうすれば、ライセンスを受けていない Genesys 製品を不用意に使用して余計な出費をこうむることはありません。

## 商標

Genesys、Genesys のロゴ、および T-Server は、Genesys Telecommunications Laboratories, Inc. の登録商標です。このマニュアルで参照されている他の商標および企業名はすべて、他社が所有権を有します。Crystal モノスペース フォントは、Software Renovation Corporation ([www.SoftwareRenovation.com](http://www.SoftwareRenovation.com)) の許可のもと使用されています。

## VAR が提供するテクニカル サポート

VAR (value-added reseller: 付加価値再販業者) からサポート契約を購入した場合、テクニカル サポートについては当該 VAR にお問い合わせください。

## Genesys が提供するテクニカル サポート

Genesys から直接サポート契約を購入した場合、以下の各地域の Genesys テクニカル サポート窓口までお問い合わせください。

地域	電話	電子メール
北米、および中南米	+888-369-5555 または +506-674-6767	<a href="mailto:support@genesyslab.com">support@genesyslab.com</a>
欧州、中東、アフリカ	+44-(0)-1276-45-7002	<a href="mailto:support@genesyslab.co.uk">support@genesyslab.co.uk</a>
アジア太平洋	+61-7-3368-6868	<a href="mailto:support@genesyslab.com.au">support@genesyslab.com.au</a>
日本	+81-3-6361-8950	<a href="mailto:support@genesyslab.co.jp">support@genesyslab.co.jp</a>

テクニカル サポート窓口にお問い合わせいただく前に、連絡先の情報、および手順全般について [『Genesys Technical Support Guide』](#) を参照してください。

## 注文、およびライセンス情報

Genesys 製品の注文、およびライセンスに関する情報全般については、[『Genesys 7 Licensing Guide』](#) を参照してください。

## 発行元

Genesys Telecommunications Laboratories, Inc. [www.genesyslab.com](http://www.genesyslab.com)

マニュアルバージョン: 76icon\_us\_05-2008\_v7.6.001.00



# 目次

<b>本書について</b>	<b>9</b>
対象読者	9
章の要約	10
パート I —Interaction Concentrator データの使用	10
パート II —Interaction Concentrator の管理	11
表記の規則	12
関連資料	13
このマニュアルに関するご意見	15
<b>パート I</b>	<b>パート I : Interaction Concentrator データの使用..... 17</b>
<b>第 1 章</b>	<b>製品の概要 ..... 19</b>
基本的アーキテクチャ	19
コンポーネントと機能	20
ICON サーバ	20
Interaction Database	22
データ ソース	22
サポートされる機能	23
Interaction Concentrator 7.6 の新機能	25
<b>第 2 章</b>	<b>IDB スキーマについて ..... 27</b>
操作テーブル	27
呼関連およびパーティ関連テーブル	28
エージェント状態関連およびログイン セッション関連テーブル	31
カスタム状態関連テーブル	32
添付データ関連テーブル	33
アウトバウンド関連テーブル	34
仮想キュー テーブル	34
構成テーブル	34
オブジェクト テーブル	34
オブジェクト リンク テーブル	35

	サービスおよび辞書テーブル.....	35
<b>第 3 章</b>	<b>Genesys Multimedia および Open Media との統合 .....</b>	<b>37</b>
	マルチメディア オブジェクト .....	37
	エンドポイント .....	38
	DN .....	38
	マルチメディア インタラクションのデータ処理と格納 .....	38
	マルチメディア レポート プロトコル イベント.....	39
	マルチメディア インタラクション.....	39
	サポートされるシナリオ .....	40
	オープン メディア インタラクションの処理 .....	41
	isOnline チャット属性 .....	41
	isOnline 属性の処理.....	42
	構成に関する推奨事項.....	43
	ICON アプリケーション.....	43
	マルチメディア用の設定.....	44
	オープン メディア用の構成 .....	44
<b>第 4 章</b>	<b>添付データの処理 .....</b>	<b>45</b>
	ボイス コールの添付データ処理 .....	45
	T-Server インタラクション .....	45
	呼固有のデータ .....	46
	履歴データ.....	46
	音声に添付されたデータのデータベース スキーマ拡張 .....	47
	添付データのセキュリティ .....	47
	マルチサイト環境と分散環境.....	47
	カスタム添付データの処理.....	48
	カスタム ディスパッチャ .....	48
	構成に関する推奨事項.....	49
	添付データ仕様ファイル .....	50
	解析の制限.....	50
	スキーマ定義 .....	51
	属性値.....	53
	基本添付データ仕様のサンプル .....	56
	カスタム添付 データの仕様のサンプル.....	57
	マルチメディア用添付データの処理 .....	58
	Interaction Server のインタラクション.....	58
	マルチメディア インタラクション固有のデータ .....	58
	マルチメディア添付データのデータベース スキーマ拡張.....	58

	マルチメディア用の添付データ仕様ファイル.....	59
	オープン メディア用の添付データ処理.....	60
	オープン メディア インタラクション固有のデータ .....	60
<b>第 5 章</b>	<b>仮想キューとルーティング ポイントのモニタ .....</b>	<b>61</b>
	仮想キューのルート結果のモニタ.....	61
	格納モード.....	62
	データ処理ステップ.....	62
	ルーティング ポイントのルート結果のモニタ.....	64
	構成に関する推奨事項.....	66
	Universal Routing Server.....	66
	ICON アプリケーション.....	67
	仮想キュー DN .....	67
	交換機.....	68
<b>第 6 章</b>	<b>エージェント状態とログインセッション .....</b>	<b>69</b>
	エージェント モデルとログイン セッション モデル.....	69
	エージェント オブジェクトとログイン セッション オブジェクト.....	69
	エージェント状態モデル — 音声およびマルチメディア .....	70
	エージェント状態モデル — SIP チャット .....	74
	ログイン セッション モデル .....	74
	入手可能なエージェント状態およびログイン セッションのデータ .....	75
	エージェント状態およびログイン セッションのデータ .....	76
	事前計算されたエージェント状態メトリクス.....	76
	後処理と準備中のエージェント状態 .....	77
	中断のない ACW または準備中の持続時間.....	77
	中断のある ACW または準備中の持続時間.....	77
	ACW とインタラクションの関連付け.....	78
<b>第 7 章</b>	<b>Outbound Contact との統合 .....</b>	<b>79</b>
	Outbound オブジェクトとモデル.....	79
	Outbound Contact オブジェクト.....	79
	キャンペーン モデル.....	80
	チェーン モデル.....	81
	入手可能な Outbound データ .....	82
	ICON と OCS の通信.....	82
	Outbound オブジェクト データ .....	82
	事前計算された OCS メトリクス.....	83
	Outbound Contact の構築シナリオ .....	84

	構成に関する推奨事項.....	87
	Outbound Contact .....	87
	ICON アプリケーション .....	88
	マルチテナント環境 .....	89
<b>第 8 章</b>	<b>ユーザ イベントとカスタム定義の状態の処理 .....</b>	<b>91</b>
	Interaction Concentrator のカスタム状態 .....	91
	EventUserEvent に含まれるデータの格納 .....	92
	カスタム状態の使用 .....	92
	エージェント デスクトップ アプリケーションの構成 .....	93
<b>パート II</b>	<b>パート II : Interaction Concentrator の管理 .....</b>	<b>95</b>
<b>第 9 章</b>	<b>Interaction Concentrator の起動と停止 .....</b>	<b>97</b>
	概要.....	97
	コマンド ライン パラメータ.....	98
	ICON の起動 .....	98
	Solution Control Interface を使用した ICON の起動 .....	99
	手動による ICON の起動.....	100
	ICON の起動 (Windows の場合) .....	101
	Windows のサービスとしての ICON の起動 .....	102
	ICON の停止 .....	103
	Solution Control Interface を使用した ICON の停止 .....	103
	ICON の停止 (UNIX の場合) .....	104
	ICON の停止 (Windows の場合) .....	105
	Windows のサービスとしての ICON の停止 .....	105
<b>第 10 章</b>	<b>Interaction Concentrator のモニタ .....</b>	<b>107</b>
	パフォーマンス カウンタへのアクセス.....	107
<b>第 11 章</b>	<b>IDB データのフィルタリング .....</b>	<b>109</b>
	概要.....	109
	フィルタリング可能なデータ .....	110
	パーティ履歴データ .....	110
	パーティメトリクス .....	110
	外部パーティ .....	111
	ユーザ データ履歴.....	111
	呼のメトリクス .....	111

	呼の履歴 .....	112
	インタラクション レコードの履歴 .....	112
	エージェント アクティビティ データ .....	112
	Service Observer データ .....	114
<b>第 12 章</b>	<b>Interaction Concentrator の HA の実装 .....</b>	<b>115</b>
	概要 .....	116
	ICON の HA モデル .....	116
	HA IDB ペアからのデータ抽出 .....	117
	リアルタイム インタラクション データ .....	117
	アウトバウンド データ .....	121
	構成データの HA .....	122
	エージェント固有データの HA .....	122
	サーバの切断 .....	123
	T-Server の切断 .....	123
	CTI リンクの切断 .....	123
	HA ペアのアクティブな T-Server の障害 .....	123
	Interaction Server の切断 .....	124
	OCS の切断 .....	124
	ICON サーバの障害 .....	124
	構成上の考慮点 .....	124
<b>第 13 章</b>	<b>構成の変更の再同期 .....</b>	<b>127</b>
	ICON が構成データを収集する方法 .....	128
	起動時の Configuration Database の読み取り .....	128
	永続キャッシュを使用できない場合 .....	129
	ICON が動的通知を受信した場合 .....	129
	ICON が構成履歴ログを読み取る場合 .....	129
	再同期のユーザ要求 .....	130
	再同期の動作 .....	130
	再同期に必要な時間 .....	132
	IDB の再同期が必要な状況 .....	132
	アラーム条件の設定 .....	133
	構成データの再同期の方法 .....	135
	再同期の推奨事項 .....	136
	バックアップを使用した Configuration Database の復元 .....	137
	Configuration Database の変更 .....	137

<b>第 14 章</b>	<b>特別なストアド プロシージャの使用 .....</b>	<b>141</b>
	マージストアド プロシージャ .....	141
	gsysIRMerge と gsysIRMerge2 .....	142
	マージ プロシージャの設定 .....	146
	gsysIRMerge2 パラメータ .....	148
	マージ プロシージャの実行 .....	149
	ページストアド プロシージャ .....	152
	ページストアド プロシージャ .....	153
	ログとエラー処理 .....	156
	ページ プロシージャのスケジューリング .....	157
	ページ プロシージャの設定 .....	158
	ページ プロシージャの実行 .....	160
	gsysInnitTimeCode スストアド プロシージャ .....	161
	時間設定プロシージャの設定 .....	162
	時間設定プロシージャの実行 .....	162
	カスタム ディスパッチャ .....	162
<b>第 15 章</b>	<b>ICON システムと構築に関するトラブルシューティング .....</b>	<b>163</b>
	起動の問題 .....	163
	Configuration Server に接続できない .....	163
	起動の際に ICON が終了する .....	164
	ランタイムの問題 .....	165
	T-Server または Interaction Server に接続できない .....	166
	ICON が T-Server から呼関連イベントを受信できない .....	166
	ICON がデータベースに情報を書き込むことができない .....	167
	ICON と Configuration Database の同期が失われる .....	168
	マージ プロシージャの問題 .....	168
	マージ プロシージャが正常に完了しない .....	169
	マージ プロシージャを実行できない .....	170
	マージ プロシージャの実行に時間がかかるまたは安定しない .....	171
<b>索引</b>	<b>.....</b>	<b>173</b>



## 本書について

『Interaction Concentrator 7.6 User's Guide』をご利用いただき、ありがとうございます。このマニュアルでは、Interaction Concentratorの基本的アーキテクチャの概要を取り上げ、さらにInteraction Concentratorの機能について詳しく説明します。また、Interaction Concentrator 7.6の継続的な保守と管理を行うために必要な情報も記載されています。

このマニュアルは、同製品のリリース7.6に対してのみ有効です。

ここでは、このマニュアルの概要、対象読者、表記規則、および参考資料について説明します。以下の項があります。

- [対象読者 \(9 ページ\)](#)
- [章の要約 \(10 ページ\)](#)
- [表記の規則 \(12 ページ\)](#)
- [関連資料 \(13 ページ\)](#)
- [このマニュアルに関するご意見 \(15 ページ\)](#)

Interaction Concentratorを使用して、Genesys Frameworkを使用する顧客インタラクションネットワーク(コンタクトセンター、エンタープライズ規模、またはマルチエンタープライズのテレフォニーおよびコンピュータネットワーク)において、インタラクションとリソースについての詳細なデータを収集し保存することができます。下流のレポートシステムがほぼリアルタイムで、Interaction Concentratorのデータにアクセスできます。

---

## 対象読者

このマニュアルは、主として、以下の項目について基本的な知識を持つシステム管理者とデータベース構築者を対象としています。

- CTI (Computer-Telephony Integration : コンピュータとテレフォニーの統合) に関する概念、処理、用語、アプリケーション
- ネットワークの設計と運用
- データベースの設計と運用
- 所有ネットワーク構成

さらに、以下の項目についての熟知する必要があります。

- Genesys Framework のアーキテクチャと機能

- コンタクトセンターで構築されている Genesys ソリューション
- リアルタイム レポートと履歴レポートの目的

---

## 章の要約

Interaction Concentratorの運用を制御するために必要な情報と機能に関する概念的な情報を区別するために、このマニュアルは大きく2つに分かれています。

### パート I —Interaction Concentrator データの使用

このマニュアルのパート I 「Interaction Concentrator データの使用」は、第1章から第8章までです。これらの章では、Interaction Concentratorのデータや機能について詳しく説明します。さらに、Interaction Concentratorの基本的アーキテクチャとコンポーネントの概要についても取り上げます。

ICON (Interaction Concentrator)の保守や機能の一部の構成を行った経験がない場合は、必ずパート I を参照してください。また、Interaction Concentratorの使用に慣れている場合も、以前の製品から変更された機能を確認するために、パート I の章に目を通すことをお奨めします。

「本書について」に加えて、パート I には以下の章があります。

- [19 ページ](#)の第 1 章「製品の概要」では、Interaction Concentrator の基本的アーキテクチャとコンポーネントについて説明します。さらに、Interaction Concentrator の機能の概要も説明します。
- [27 ページ](#)の第 2 章「IDB スキーマについて」では、IDB アーキテクチャの概要を説明します。さらに、統計テーブルで使用可能な事前に計算された呼とパーティメトリクスの概要も説明します。
- [37 ページ](#)の第 3 章「Genesys Multimedia および Open Media との統合」では、マルチメディア インタラクションとオープン メディア インタラクションについてのデータを、IDB で使用可能にする方法を説明します。
- [45 ページ](#)の第 4 章「添付データの処理」では、ボイス コールと Multimedia インタラクションに添付されたユーザデータを、IDB で使用可能にする方法を説明します。この章には、T-Server と Interaction Server が提供するイベント レポートの添付データに含まれる KVP (Key-Value Pairs : キーと値のペア)を処理するための XML (Extensible Markup Language : 拡張マークアップ言語) スキーマの定義が記載されています。必要な添付データの仕様のサンプルも記載されています。
- [61 ページ](#)の第 5 章「仮想キューとルーティング ポイントのモニタ」では、Interaction Concentrator がどのように仮想キューとルーティング ポイントをモニタし、ルーティング情報を Interaction Database (IDB) に格納するかを説明します。
- [69 ページ](#)の第 6 章「エージェント状態とログインセッション」では、Interaction Concentrator で使用されるエージェント状態とログインセッションモデルについて説明します。エージェント状態とログインセッションのデータを IDB で使用可能にする方法も説明します。

- [79 ページ](#)の第 7 章「Outbound Contact との統合」では、Genesys Outbound Contact ソリューションのデータを IDB で使用可能にする方法を説明します。
- [91 ページ](#)の第 8 章「ユーザ イベントとカスタム定義の状態の処理」では、顧客が定義したエージェント状態についてのデータおよびボイス コールに添付された一般データを、IDB で使用可能にする方法を説明します。

## パート II —Interaction Concentrator の管理

このマニュアルのパート II 「Interaction Concentrator の管理」は、第 9 章から第 15 章までです。Interaction Concentrator の継続的な保守と管理を行うために、すべてのユーザが必要とする情報が記載されています。たとえば、ICON の起動、停止、モニタの方法、HA (High Availability : 高可用性) やデータ フィルタリングなどさまざまな機能の構成方法について知ることができます。さらに、起動時とランタイムの問題のトラブルシューティングやオンデマンドで Interaction Database (IDB) を同期する方法など、重要な情報が記載されています。

このマニュアルのパート II には以下の章があります。

- [97 ページ](#)の第 9 章「Interaction Concentrator の起動と停止」では、Interaction Concentrator の起動の前提条件および ICON の起動と停止の手順について説明します。
- [107 ページ](#)の第 10 章「Interaction Concentrator のモニタ」では、HTTP Listener にアクセスして Interaction Concentrator のパフォーマンスをモニタしてレポートする方法を説明します。
- [109 ページ](#)の第 11 章「IDB データのフィルタリング」では、Interaction Concentrator 7.6 のデータ フィルタリング機能と、オプションを設定してこの機能を構成する方法を説明します。
- [115 ページ](#)の第 12 章「Interaction Concentrator の HA の実装」では、Interaction Concentrator の HA の設計と実装について説明します。
- [127 ページ](#)の第 13 章「構成の変更の再同期」では、IDB データを強制的に再同期する必要がある状況と、再同期を実行する方法について説明します。
- [141 ページ](#)の第 14 章「特別なストアードプロシージャの使用」では、ICON ユーザに特に関係があるストアードプロシージャと、これを設定して実行する手順について説明します。
- [163 ページ](#)の第 15 章「ICON システムと構築に関するトラブルシューティング」では、Interaction Concentrator の起動時とランタイムに発生するよくある問題の解決方法を取り上げます。

## 表記の規則

このマニュアルでは、特定の種類の情報を簡単に識別できるように、いくつかの表記上の規則を採用しています。

### マニュアルのバージョン番号

このマニュアルの表紙の裏側の下部にバージョン番号が記載されています。マニュアルに新しい情報が追加されると、バージョン番号が変更されます。バージョン番号の例を以下に示します。

76icon\_dep\_05-2008\_v7.6.001.00

この番号は、製品について Genesys テクニカル サポートに問い合わせる際に必要です。

### 字体スタイル

#### 斜体

このマニュアルでは、強調、マニュアルのタイトル、専門的な用語の定義（または初出の専門用語）、および数学的な変数を表すために斜体を使用されます。

- 例：
- 詳細については、『*Genesys 7 Migration Guide*』を参照してください。
  - *慣例*および*恒例*は、特定の業界や職業で一般的に認知され実施される行為を指します。
  - このオプションには、この値を使用しないでください。
  - 数式  $x+1=7$  で、 $x$  が表すのは、...

#### モノスペース フォント

teletype や typewriter text のように表記されるモノスペース フォントは、プログラミングの識別子や GUI 要素を表す目的で使用されます。

これには、ディレクトリ、ファイル、フォルダ、構成オブジェクト、パス、スクリプト、ダイアログ ボックス、オプション、フィールド、テキスト ボックス、リスト ボックス、操作モード、ラジオ ボタンを含むすべてのボタン、チェック ボックス、コマンド、タブ、CTI イベント、エラー メッセージの名前、オプションの値、論理引数、コマンド構文、コードの例などが含まれます。

- 例：
- [Show variables on screen] チェック ボックスをオンにします。
  - [Summation] ボタンをクリックします。
  - [プロパティ] ダイアログ ボックスに、ご使用の環境のホスト サーバの値を入力します。
  - [Operand] テキスト ボックスに数式を入力します。
  - [OK] をクリックして、[プロパティ] ダイアログ ボックスを閉じます。

- 以下の表に、EventError イベントの発生時に T-Server® が送信する全エラーメッセージを示します。
- inbound-bsns-calls オプションに真を選択した場合、ローカル エージェントで確立されたインバウンド通話はすべてビジネス通話として認識されます。

モノスペース フォントは、構成やインストールの際やコマンドラインに、ユーザが手動で入力する必要があるテキストにも使用されます。

例： 

- コマンドラインに exit と入力します。

## このマニュアルで使用される画面キャプチャ

このマニュアルで使用される製品の GUI (Graphical User Interface : グラフィカル ユーザ インタフェース) の画面キャプチャには、スペル、大文字の使い方、文法などに小さな誤りが含まれている場合があります。訂正によって製品のインストール、構成、および通常の利用に問題が生じる場合を除いて、画面キャプチャにテキストを付記する形で訂正しています。たとえば、アイコン名に慣用上の誤りが含まれている場合でも、名前が製品 GUI にそのまま表示される場合は、付随するテキストで誤りは訂正されていません。

## 角かっこ

角かっこは、論理引数、コマンド、プログラミング構文に含まれるオプションのパラメータや変数を表します。これらのパラメータや値は、引数、コマンド、またはコードブロックを処理するために必須ではありません。このオプション情報を使用するかどうかは、ユーザが判断します。以下に例を示します。

```
smcp_server -host [/flags]
```

## 山形かっこ

山形かっこは、ユーザが指定する必要がある値のプレースホルダを表します。エンタープライズ環境に固有の DN またはポート番号などが該当します。以下に例を示します。

```
smcp_server -host <confighost>
```

---

## 関連資料

必要に応じて、以下の資料も参照してください。

- 『Interaction Concentrator 7.6 Deployment Guide』。Interaction Concentrator のインストールと構成に役立ちます。
- 『Interaction Concentrator 7.6 Physical Data Model』。ご使用の RDBMS (Relational Database Management System : 関係型データベース管理システム) のタイプに対応する IDB テーブルについて知ることができます。

- 『Interaction Concentrator 7.6 Database Size Estimator』。構築計画の際に、IDBのサイズを見積もるために役立ちます。Estimator は Microsoft Excel スプレッドシートで、Genesys テクニカル サポートの Web サイトで入手できます。
- 『Genesys 7 Hardware Sizing Guide』。推奨ハードウェア アーキテクチャについての情報と、データベース サイズの見積もりに関する追加情報が記載されています。
- Genesys 7.6 Combined Log Events のヘルプ。Interaction Concentrator を含む Genesys サーバ アプリケーションが生成するログ イベントが記載されています。
- Genesys Info Mart リリース 7.6 のマニュアルセット。Interaction Concentrator を Genesys Info Mart のデータ ソースとして使用する場合に役立ちます。
- Genesys Outbound Contact リリース 7.6 のマニュアルセット。アウトバウンド関連のデータを IDB に格納する場合に役立ちます。
- Genesys Universal Routing リリース 7.6 のマニュアルセット。インタラクシオン処理の仮想キュー使用に関する情報を IDB に格納する場合に役立ちます。
- Genesys Multimedia リリース 7.6 のマニュアルセット。Multimedia インタラクシオンについてのインタラクシオン関連データとエージェント関連データを、IDB に格納する場合に役立ちます。
- 『Genesys Technical Publications Glossary』 (Genesys Documentation Library DVD に収録)。このマニュアルで使用される Genesys および CTI の用語と頭字語の総合的なリストが含まれています。
- 『Genesys 7 Migration Guide』 (Genesys Documentation Library DVD に収録)。Genesys 製品のリリース 6.x 以上から Genesys 7.x リリース全般への移行ストラテジが記載されています。詳細については、Genesys テクニカル サポートにお問い合わせください。
- この製品の Release Notes (リリース ノート) および Product Advisories (製品に関するアドバイス) は、Genesys テクニカル サポートの Web サイト (<http://genesyslab.com/support>) で入手できます。

サポートされるハードウェアとサードパーティ ソフトウェアについては、Genesys テクニカル サポートの Web サイトで以下のマニュアルを参照してください。

- 『Genesys 7 Supported Operating Systems and Databases』
- 『Genesys 7 Supported Media Interfaces』

Genesys 製品マニュアルは、以下の場所から入手できます。

- Genesys テクニカル サポートの Web サイト (<http://genesyslab.com/support>)
- Genesys Documentation Library DVD。Genesys Order Management ([orderman@genesyslab.com](mailto:orderman@genesyslab.com)) まで電子メールでご注文ください。

---

## このマニュアルに関するご意見

このマニュアルについてのご意見・ご要望がある場合は、  
[Techpubs.webadmin@genesyslab.com](mailto:Techpubs.webadmin@genesyslab.com)まで電子メールでコメントをお寄せください。

記載内容の誤りや欠落、不正確な点、構成、取り扱うテーマ、範囲など、このマニュアルに関するご意見をお待ちしています。ご意見の内容については、このマニュアルに記載された内容とその記載要領に限定させていただきます。製品自体に関するご意見については、Genesysテクニカルサポートまでご連絡ください。

お客様からご意見をお寄せいただいた時点で、そのご意見については、お客様に対する一切の義務を負うことなく、Genesysが適切であるとみなす任意の方法で使用、および配布する非排他的権利がGenesysに与えられるものとします。





パート

# パート I : Interaction Concentrator データの使用

このマニュアルのパート I では、Interaction Concentratorのデータや機能について詳しく説明します。さらに、Interaction Concentratorの基本的アーキテクチャとコンポーネントの概要についても取り上げます。

パート I には以下の章があります。

- [19 ページ](#)の第 1 章「製品の概要」
- [27 ページ](#)の第 2 章「IDB スキーマについて」
- [37 ページ](#)の第 3 章「Genesys Multimedia および Open Media との統合」
- [45 ページ](#)の第 4 章「添付データの処理」
- [61 ページ](#)の第 5 章「仮想キューとルーティング ポイントのモニタ」
- [69 ページ](#)の第 6 章「エージェント状態とログインセッション」
- [79 ページ](#)の第 7 章「Outbound Contact との統合」
- [91 ページ](#)の第 8 章「ユーザ イベントとカスタム定義の状態の処理」





# 1

## 製品の概要

この章では、Interaction Concentrator の基本的アーキテクチャとコンポーネントについて説明します。さらに、Interaction Concentrator の機能の概要も説明します。以下の項があります。

- [基本的アーキテクチャ \(19 ページ\)](#)
- [コンポーネントと機能 \(20 ページ\)](#)
- [サポートされる機能 \(23 ページ\)](#)
- [Interaction Concentrator 7.6 の新機能 \(25 ページ\)](#)

---

## 基本的アーキテクチャ

Interaction Concentrator は、Genesys ソフトウェアを使用するコンタクトセンターのさまざまなソースから、詳細なデータを収集して保管するための Genesys 製品です。下流のレポートシステムがほぼリアルタイムで、Interaction Concentrator のデータにアクセスできます。

Interaction Concentrator 製品は Genesys Framework の上位で動作し、Interaction Concentrator (ICON) と呼ばれるサーバアプリケーションと Interaction Database (IDB) と呼ばれるデータベースから構成されます。Configuration Server、T-Server、または特定の Genesys ソリューションなどのデータソースからサーバがデータを受け取り、Genesys DB Server を介して IDB にデータを格納します。

[図 1](#) は ICON の基本的アーキテクチャを示しています。わかりやすいように Framework コンポーネントが省略されています。

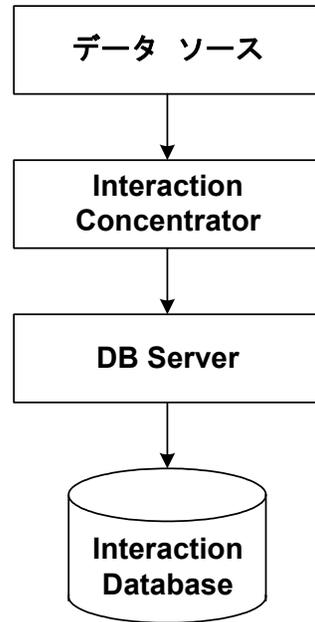


図 1: Interaction Concentrator の基本的アーキテクチャ

## コンポーネントと機能

Interaction Concentrator は以下の要素から構成されています。

- ICON サーバ
- Interaction Database

以下の項でそれぞれの要素について説明します。

### ICON サーバ

ICON サーバは以下のような処理を行います。

- Configuration Server、T-Server、Interaction Server、および Outbound Contact Server から受け取ったイベントの再処理。ICON インスタンスに構成された役割に従って再処理します。

処理はインメモリ キュー (アキュムレータ) で実行されます。インメモリ キューのサイズ、またはインメモリ キューから永続キューにデータを書き込む間隔を設定できます。インメモリ キューの構成オプションの詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のオプションの構成についての章を参照してください。

- IDB に格納するデータの準備。
- 準備したデータをインメモリ キューから永続キューに書き込みます。永続キューの詳細については、「[永続キュー](#)」を参照してください。
- 永続キューのデータの管理。
- 永続キューから IDB へのデータ書き込み。

ICONの機能とパフォーマンスを決定する構成オプションの詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』を参照してください。

## 永続キュー

永続キューはICONが作成するファイルで、IDBに書き込む前のデータを格納します。また、IDBに対するデータベース書き込み要求についての情報も格納されます。永続キューのデータはICONをシャットダウンして再起動しても保持されます。

### Configuration Database の同期

ICON 7.5 の場合は、永続キューが1つあり、ICONのすべてのインスタンスと役割のデータが格納されます。リリース7.6の場合は、`cfg`の役割を実行するICONインスタンス用に追加の永続キュー(`cfg-sync.db`)があります。このキューが、IDBとConfiguration Databaseの同期を維持するために重要な役割を果たします。ICONが構成データ変更のタイムスタンプを永続キューに格納し、起動時にConfiguration Serverから、そのタイムスタンプ以降に発生したすべての構成変更を要求します。同期を維持するための永続キューの役割の詳細については、[128ページ](#)の「ICONが構成データを収集する方法」を参照してください。

### 永続キューの構成オプション

ICON の構成オプションで、以下の項目を指定できます。

- すべての役割の永続キューのファイル名(`cfg`の役割を除く)。ICONインスタンスの役割が `cfg` の永続キューの名前は `cfg-sync.db` で、変更はできません。
- ICONが永続キューからデータを消去する頻度(コミットされるトランザクションの数)。
- 環境の障害アラームのしきい値。
- 起動時の永続キューの動作。

アラームしきい値は、ICONのパフォーマンスのモニタにも使用できます。永続キューの構成オプションの詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』を参照してください。

---

**注：** ICON には永続キューの正式なサイズ制限はありませんが、オペレーティングシステムによる制限が適用される場合があります。

---

## ICON サーバのインタフェース

ICON サーバは以下のようなインタフェースを提供します。

- Local Control Agent (LCA) を介した Solution Control Server (SCS) とのインタフェース。ICON サーバの起動と停止を制御します。
- Configuration Server とのインタフェース。Interaction Concentrator アプリケーションの構成オプション、および Interaction Concentrator の機能に関連する他の構成オブジェクトとオプションを読み取ります。このインタフェースは、ICON と Configuration Server との接続 (コンタクトセンターリソースに関するデータのデータソースとして)とは論理的に分離されています([22ページ](#)の「データソース」を参照)。

- Message Server とのインタフェース。Central Logger にメッセージのログを記録するためです。

**注：** Interaction Concentrator では、ICON がインタフェースを提供するコンポーネント間の安全なデータ交換のために、TLS (Transport Layer Security : トランスポート層セキュリティ) プロトコルはサポートされません。

## Interaction Database

Interaction Database には、コンタクトセンター インタラクションとリソースに関するデータが、詳細なレベルで格納されます。IDB は格納(データ挿入)のために最適化されたデータベースです。Interaction Concentrator 自体にはレポート機能はありません。下流のレポートアプリケーションのために、IDB を一貫性のある信頼性の高いデータ ソースとして使用できます。

IDB アーキテクチャの概要については、27 ページの「IDB スキーマについて」を参照してください。IDB のすべてのテーブルとフィールドの完全なテーブル構造と説明については、ご使用の RDBMS に該当するものについて、『Interaction Concentrator 7.6 Physical Data Model』を参照してください。

### ストアド プロシージャ

Interaction Concentrator ではいくつかのストアド プロシージャを使用します。これらのほとんどは Interaction Concentrator の機能に完全に組み込まれており、エンドユーザには関係ありません。ただし、以下のストアド プロシージャの場合はユーザの入力やアクションが必要です。

- |                         |   |
|-------------------------|---|
| <b>マージ</b>              | <ul style="list-style-type: none"> <li>• gsysIRMerge および gsysIRMerge2— 閉じたシングルサイト インタラクションとマルチサイト インタラクションのデータ処理をファイナライズするマージ プロシージャ</li> </ul> |
| <b>パージ</b>              | <ul style="list-style-type: none"> <li>• gsysPurgeIR、gsysPurgeUDH、gsysPurgeLS、gsysPurgeOS— マージした音声インタラクションを安全に IDB から消去するパージ プロシージャ</li> </ul>  |
| <b>時間設定</b>             | <ul style="list-style-type: none"> <li>• gsysInitTimeCode— G_TIMECODE テーブルを設定して、時間間隔のレポートを有効にするストアド プロシージャ</li> </ul>                           |
| <b>カスタム<br/>ディスパッチャ</b> | <ul style="list-style-type: none"> <li>• gudCustDisp1 および gudCustDisp2— 添付データ処理をカスタマイズするストアド プロシージャ</li> </ul>                                  |

これらのストアド プロシージャの詳細については、141 ページの「特別なストアド プロシージャの使用」を参照してください。

## データ ソース

表1は、ICON がデータを収集するソースの要約を示しています。Interaction Concentrator 7.5 と 7.6 は、Interaction Concentrator 7.2 よりも広範なデータ タイプとソースに対応します。

表 1: ICON のデータ ソース

データのタイプ	ソース	対応するInteraction Concentratorのリリース		
		7.2	7.5	7.6
コンタクト センター リソースの構成データ	Configuration Server	○	○	○
コンタクトセンターの呼アクティビティに関するCTI関連の詳細なデータ	T-Server	○	○	○
コンタクトセンターのVOIP (Voice over IP)インタラクションアクティビティに関するCTI関連の詳細なデータ	T-Server (SIP Server)	×	×	○
コンタクトセンターのオープンメディアインタラクションアクティビティに関する詳細なデータ(電子メールとSIP以外のチャットを含む)	Interaction Server	×	○	○
コンタクトセンターのSIPチャットインタラクションアクティビティに関する詳細なデータ	SIP Server	×	○	○
オープン(カスタム定義)メディアのメディアタイプ	Interaction Server	×	×	○
インタラクション処理の仮想キュー使用に関する詳細なデータ	<ul style="list-style-type: none"> <li>ボイス コールの場合：T-Server (Universal Routing Server [URS] から)</li> <li>マルチメディア インタラクションの場合： Interaction Server (URS から)</li> </ul>	×	○	○
アウトバウンド呼およびキャンペーン固有のデータ	OCS	○	○	○

## サポートされる機能

Interaction Concentrator は、[25 ページ](#)の「Interaction Concentrator 7.6 の新機能」で説明した新機能に加え、Interaction Concentrator インスタンスに構成した役割に応じて、コンタクトセンターアクティビティに関するレポートをサポートする以下の機能を提供します。

- 現在のコンタクトセンターの構成(オブジェクトと関連付け)に関する情報の収集と保存、削除された構成オブジェクトと終了した関連付けに関する情報の保持。

- アクティブな音声インタラクションと終了した音声インタラクションに関する詳細情報の収集と保存 ( 交換機、DN、時刻、呼とパーティに関するルーティング情報を含む )。Interaction Concentrator ではグローバルに一意の呼 ID を使用。
- マルチメディア インタラクション(電子メールとチャット)に関する詳細情報の収集と保存。詳細については、[37 ページ](#)の第 3 章を参照してください。
- 音声インタラクションとマルチメディア(電子メールとチャット)インタラクションを処理するエージェントの状態とログインセッションに関する詳細情報の収集と保存。詳細については、[69 ページ](#)の第 6 章を参照してください。
- カスタム エージェント状態のサポート ( 音声インタラクションを処理するエージェントの場合 )。詳細については、[69 ページ](#)の第 6 章を参照してください。
- インタラクション処理における仮想キュー使用に関する詳細情報の収集と保存 ( 全タイプのインタラクションについて )。詳細については、[61 ページ](#)の第 5 章を参照してください。
- 呼処理におけるルーティング ポイント (RP) 使用に関する詳細情報の収集と保存 ( 音声インタラクションの場合 )。
- ネットワーク ベースの Contact Solution 環境で生成されるインタラクションに関する詳細情報の収集と保存。
- ネットワーク コール パーキング環境で生成されるインタラクションに関する詳細情報の収集と保存。
- 音声インタラクションとマルチメディア インタラクション ( 電子メールとチャット ) の添付データの保存、および添付データ変更の履歴の収集。詳細については、[45 ページ](#)の第 4 章を参照してください。
- ボイス コールのカスタム添付データ処理のサポート。詳細については、[45 ページ](#)の第 4 章を参照してください。
- 以下のアウトバウンド キャンペーンに関する詳細情報の収集と保存。
  - ◆ キャンペーン処理の履歴
  - ◆ チェーン処理の履歴
  - ◆ OCS が提供する事前に計算されたメトリクス詳細については、[79 ページ](#)の第 7 章を参照してください。
- リアルタイムの同一日内レポートをサポート。インタラクション完了後ではなく、データが入手可能になると即座に IDB にデータが書き込まれます。
- IS-Links (Inter-Site Call Linkage) を使用し、マルチサイト インタラクションを処理する高度な認識メカニズムを提供。ICON が T-Server から、所定のインタラクションと別のサイトのインタラクションの関係に関する情報を受け取ります。これにより、サイト間のレポートに完全なデータを使用できます。Interaction Concentrator がストアドプロシージャを提供して、マルチサイト インタラクションのインタラクション レコードをマージします。マージプロシージャの詳細については、[141 ページ](#)の第 14 章「特別なストアドプロシージャの使用」を参照してください。

- マルチバイト文字のエンコードをサポート。
- 以下の 2 つのフォーマットで時刻情報を格納。
  - ◆ GMT (Greenwich Mean Time : グリニッジ平均時)—datetime データ型として。
  - ◆ 秒単位の UTC (Coordinated Universal Time : 協定世界時)—integer データ型として。

ICON はデータプロバイダ イベント (たとえば、T-Server の TEvents) のタイムスタンプから、UTC (秒単位) の形式で時刻情報を取得します。

## Interaction Concentrator 7.6 の新機能

Interaction Concentrator リリース 7.6 で追加された機能や変更された機能を以下に示します。

- オンデマンドで、Interaction Database (IDB) の構成データを Configuration Database と再同期可能。詳細については、[127 ページ](#)の第 13 章を参照してください。
- 特定タイプのデータのフィルタリング メカニズムを提供し、データベースサイズの最適を実現。詳細については、[109 ページ](#)の第 11 章を参照してください。
- 複数の IDB を使用してデータ冗長性を実現し、ETL (Extraction, Transformation and Loading : 抽出、変換およびロード) レベルで、データの高可用性をサポート。適用されるデータは、構成データ、エージェント関連データ、音声詳細データおよび SIP チャット データです。詳細については、[115 ページ](#)の第 12 章を参照してください。
- ACW (After-Call Work : 後処理) と関連付けられた最初のインタラクションの後処理をレポート可能。詳細については、[69 ページ](#)の第 6 章を参照してください。
- 以前に電子メールと非 SIP データがサポートされていたすべての領域で、オープンメディアのメディア タイプをサポート。詳細については、[37 ページ](#)の第 3 章を参照してください。
- SIP チャットのレポートをサポート、チャットセッションにフォーカスがあるかどうかを識別可能。詳細については、[37 ページ](#)の第 3 章および [69 ページ](#)の第 6 章「エージェント状態とログインセッション」を参照してください。
- エージェントが受信するインタラクションまたはエージェントが処理するインタラクションによる後処理 (ACW) および NotReady エージェント状態の中断を抑制可能。詳細については、[69 ページ](#)の第 6 章を参照してください。





# 2

## IDB スキーマについて

Interaction Databaseは、ICONが提供するすべてのレポートデータをテーブルに格納します。論理的には、ICONが書き込む情報のクラスに応じて、IDBテーブルをグループに分割できます。

この章ではテーブルの論理グループについて説明します。さらに、統計テーブルで使用可能な事前に計算された呼とパーティメトリクスの概要も説明します。以下の項があります。

- [操作テーブル \(27 ページ\)](#)
- [構成テーブル \(34 ページ\)](#)
- [サービスおよび辞書テーブル \(35 ページ\)](#)

---

**注：** IDB のすべてのテーブルとフィールドの完全なテーブル構造と説明については、ご使用のRDBMSに該当するものについて、『*Interaction Concentrator 7.6 Physical Data Model*』を参照してください。

データフィルタリング機能を使用して、IDBに格納するデータを制限できます。詳細については、[109 ページ](#)の「IDBデータのフィルタリング」を参照してください。

---

---

## 操作テーブル

操作データを格納するテーブルには、以下の6つのサブグループがあります。

- 呼関連およびパーティ関連テーブル
- エージェント状態関連およびログインセッション関連テーブル
- カスタム状態関連テーブル
- 添付データ関連テーブル
- アウトバウンド関連テーブル
- 仮想キュー関連テーブル

これらのテーブルのグループについては、以下のサブセクションで詳しく説明します。

## 呼関連およびパーティ関連テーブル

呼とパーティに関するデータを格納するテーブルには、以下の3つのサブグループがあります。

- コア テーブル
- 履歴テーブル
- 統計テーブル

以下のサブセクションで、それぞれのサブグループについて説明します。

### コア テーブル

コア テーブルには、呼、パーティ、およびすべての関連情報について、現在(最新)の情報が格納されます。

このサブグループには以下の5つのテーブルがあります。

- G\_CALL— 電話(完了およびアクティブ)、SIP、チャット、およびオープンメディア インタラクションに関する情報を格納
- G\_PARTY— 通話パーティに関する情報を格納
- G\_IR— 特定のシナリオのすべての呼に共通のデータに関する情報を格納
- G\_IS\_LINK— マルチサイト シナリオの呼リンクに関する情報を格納
- G\_ROUTE\_RESULT— 呼のルーティング結果に関する情報を格納

### 履歴テーブル

履歴テーブルには、前にコア テーブルに格納されたデータの間(履歴)状態が格納されます。履歴テーブルの名前は、G\_CALL\_HISTORY、

G\_PARTY\_HISTORY、G\_IR\_HISTORY、およびG\_IS\_LINK\_HISTORYです。これらはそれぞれ、G\_CALL、G\_PARTY、G\_IR、およびG\_IS\_LINKコア テーブルに対応します。

### 統計テーブル

2つの統計テーブルG\_CALL\_STATとG\_PARTY\_STATには、それぞれ呼とパーティに関する統計情報が格納されます。

#### パーティ メトリクス

表2には、G\_PARTY\_STATテーブルに格納されるInteraction Concentratorのメトリクスが、テーブルに現れる順序で示されています。また、表2には、メトリクスの計算対象のメディア タイプも示されています。

表 2: 使用可能なパーティ メトリクス (メディア タイプ別)

メトリクス名	説明	サポートされる メディア タイプ		
		音声	マルチメディア	
			電子 メール	チャット
TT_ALERTING	パーティがALERTING (アラート) 状態だった合計時間。	○	○	○
TT_CONNECTED	パーティがCONNECTED (接続済み) 状態だった合計時間。呼内の他のパーティの状態はこのメトリクスに影響しない。	○	○	○
TT_HOLD	パーティがHOLD (保留中) 状態だった合計時間。	○	×	×
TT_QUEUED	パーティがQUEUED (キュー内待機) 状態だった合計時間。	○	○	○
TT_ACW	パーティが後処理(ACW)中だった合計時間。	○	×	×
CNT_ALERTING	パーティの状態がアラートに変化した回数。	○	○	○
CNT_CONNECTED	パーティの状態が接続済みに変化した回数(たとえば、アラートまたは保留中状態から)。	○	○	○
CNT_HOLD	パーティの状態が保留中に変化した回数。	○	×	×
CNT_QUEUED	パーティの状態がキュー内待機に変化した回数。	○	○	○
CNT_ACW	パーティにACWが存在することを示すフラグ。	○	×	×
TT_ON_ALERT	インタラクション内の別のパーティがALERTING (アラート) 状態だった合計時間。	○	○	○
TT_ON_HOLD	呼の別のパーティがHOLD (保留中) 状態だった合計時間。	○	×	×
TT_ON_QUEUE	インタラクション内の別のパーティがQUEUED (キュー内待機) 状態だった合計時間。	○	○	○
TT_ON_CONNECTED	呼のすべてのパーティが同時にCONNECTED (接続済み) 状態だった合計時間。	○	○	○

表2: 使用可能なパーティ メトリクス (メディア タイプ別) (つづき)

メトリクス名	説明	サポートされるメディアタイプ		
		音声	マルチメディア	
			電子メール	チャット
T_DURATION	パーティの存続時間。	○	○	○
PM_EXT_1～ PM_EXT_10	カスタム メトリクスの計算用に予約。 カスタムのストアードプロシージャを使用してカスタム メトリクスを計算可能。	○	○	○

**インタラクション  
メトリクス** 表3には、G\_CALL\_STAT テーブルに格納される Interaction Concentrator のメトリクスが、テーブルに現れる順序で示されています。また、表3には、メトリクスの計算対象のメディアタイプも示されています。

表3: 使用可能なインタラクションメトリクス (メディアタイプ別)

メトリクス名	説明	サポートされるメディアタイプ		
		音声	マルチメディア	
			電子メール	チャット
F_CONN	接続済みのフラグ。	○	○	○
F_CONN_EXTN	内線に接続済みのフラグ。	○	○	○
F_TE_ABND	イベント放棄のフラグ。	○	×	○
CNT_HOLD	呼が保留された回数。	○	×	×
CNT_DIVERT	インタラクションが転送された回数(たとえば、キューまたはルーティングポイントから)。	○	○	○
CNT_TRANSFER	インタラクションが転送された回数(転送が完了した場合)。	○	○	○
CNT_TRANSFER_LGIN	ログインしたエージェントが存在するデバイスに関連付けたパーティによってインタラクションが転送された回数。	○	○	○
CNT_CONFERENCE	インタラクションで会議通話が行われた回数。	○	×	○
T_DURATION	インタラクションの存続時間。	○	○	○

表 3: 使用可能なインタラクションメトリクス(メディアタイプ別)(つづき)

メトリクス名	説明	サポートされるメディアタイプ		
		音声	マルチメディア	
			電子メール	チャット
T_CONN	インタラクション内の最初のパーティに接続するまでの時間。 このメトリクスの値は、(エージェントまたはIVRが)任意のデバイスで応答するまでの時間だとみなされる。	○	○	○
T_CONN_EXTN	内線に接続するまでの時間。	○	○	○
T_TE_ABND	放棄されるまでの時間。	○	×	○
TT_ALERTING	ALERTING (アラート) 状態の内部パーティが少なくとも1つ呼に存在した場合のすべての時間間隔の合計。	○	○	○
TT_CONNECTED	呼のすべてのパーティがCONNECTED (接続済み) 状態だったときのすべての時間間隔の合計。	○	○	○
TT_HOLD	HOLD (保留中) 状態の内部パーティが少なくとも1つ呼に存在した場合のすべての時間間隔の合計。	○	×	×
TT_QUEUED	QUEUED (キュー内待機) 状態の内部パーティが少なくとも1つ呼に存在した場合のすべての時間間隔の合計。	○	○	○
CM_EXT_1~ CM_EXT_10	カスタムメトリクスの計算用に予約。 カスタムのストアードプロシージャを使用してカスタムメトリクスを計算可能。	○	○	○

## エージェント状態関連およびログインセッション関連テーブル

コンタクトセンターのエージェントに関する履歴データを保存するテーブルには、以下の3つのサブグループがあります。

- コアテーブル
- 履歴テーブル
- 統計テーブル

以下のサブセクションで、それぞれのサブグループについて説明します。

## コア テーブル

コア テーブルには、エージェント状態、ログインセッション、およびセッションとエンドポイント(DN)の関連付けについての情報が格納されます。

このサブグループには以下の3つのテーブルがあります。

- G\_LOGIN\_SESSION— エージェントのログイン セッションに関する情報を格納
- GX\_SESSION\_ENDPOINT—ログインセッションとエンドポイント(DN)の関連付けに関する情報を格納
- G\_AGENT\_STATE\_RC— エージェント状態の理由コードに関する情報を格納

## 履歴テーブル

履歴テーブルには、エージェント状態およびDNのログインセッションについての履歴データが格納されます。

このサブグループには以下の2つのテーブルがあります。

- G\_AGENT\_STATE\_HISTORY—所定のログインセッション内のエージェント状態の履歴を格納
- G\_DND\_HISTORY—デバイス(DN)のDND(Do Not Disturb: 着信拒否)機能のアクティブ化と非アクティブ化の履歴を格納

## 統計テーブル

2つの統計テーブルには、エージェント状態とログインセッションに関連する統計が格納されます。

このサブグループには以下の2つのテーブルがあります。

- GS\_AGENT\_STAT— エージェント状態の持続時間を格納
- GS\_AGENT\_STAT\_WM— エージェント状態の作業モードの持続時間を格納

## カスタム状態関連テーブル

Interaction Concentratorは、Call Concentratorとの互換性を保つため、UserEventsで、顧客が定義する状態と添付データをサポートします。

以下の3つのテーブルがカスタム状態に関連します。

- G\_CUSTOM\_DATA\_P—ボイス コールに関連付けたUserEventsで提供されるキー値を含むフラット テーブル
- G\_CUSTOM\_DATA\_S—汎用テーブル。キー値に関する情報はボイス コールに関連付けた UserEvents が提供
- G\_CUSTOM\_STATES— エージェントのログイン セッション中のエージェント状態の変化に関する詳細情報を格納

## 添付データ関連テーブル

添付データ(UserDataとも呼ばれる)に関連するテーブルには、T-Serverおよび(該当する場合は) Interaction Serverがインタラクションに添付するデータが格納されます。ICONがデータをTEventsまたはマルチメディア レポート イベントから選択します。ICONが選択する正確なデータは、ICONの構成によって異なります。これらのテーブルのデータには、添付データの現在の状態や添付データ変更の履歴(構成されている場合)が含まれます。

添付データ関連テーブルには、以下の3つのサブグループがあります。

- 音声用添付データ状態テーブル(フラットテーブル)
- 音声用添付データ履歴テーブル(汎用テーブル)
- マルチメディア添付データ テーブル

以下のサブセクションで、それぞれのサブグループについて説明します。詳細については、[45ページ](#)の第4章「添付データの処理」を参照してください。

### 添付データ状態テーブル

添付データ状態(フラット)テーブルには、呼に関連付けられた添付データ属性の現在の状態(受信した最新の状態)が格納されます。

このサブグループには以下の4つのテーブルがあります。

- G\_CALL\_USERDATA— 事前定義された添付データ属性を格納
- G\_CALL\_USERDATA\_CUST— カスタムの添付データ属性を格納
- G\_CALL\_USERDATA\_CUST1— カスタムの添付データ属性を格納
- G\_CALL\_USERDATA\_CUST2— カスタムの添付データ属性を格納

格納されるマルチメディア データの詳細については、[58ページ](#)の「マルチメディア添付データのデータベース スキーマ拡張」を参照してください。

### 添付データ履歴テーブル

添付データ履歴(汎用)テーブルには、添付データについての履歴情報が格納されます。

このサブグループには以下の2つのテーブルがあります。

- G\_USERDATA\_HISTORY— セキュリティ保護が不要な添付データ フィールドに関する情報を格納
- G\_SECURE\_USERDATA\_HISTORY— セキュリティ保護が必要な機密の添付データ フィールドに関する情報を格納(顧客の社会保障番号など)

## マルチメディア添付データ テーブル

マルチメディア添付データ テーブルには、マルチメディア添付データに関する情報が格納されます。

このサブグループには以下の2つのテーブルがあります。

- GM\_L\_USERDATA— 添付データ キーの値を格納 (推奨応答、自動応答、受信確認、顧客 ID、処理停止の理由)
- GM\_F\_USERDATA— 電子メールおよびチャット インタラクションに関するメタデータ情報を格納 (送信者の名前と電子メールアドレス、件名、タイプなど)

## アウトバウンド関連テーブル

アウトバウンド関連テーブルは、名前にGO\_またはGOX\_のプレフィックスが付きます。これらのテーブルには、Outbound Contact Server (OCS)がレポートするアウトバウンド呼とアウトバウンド キャンペーンの処理に関連するエンティティと属性についての情報が格納されます。

格納されるアウトバウンド データの詳細については、[82ページ](#)の「入手可能なOutboundデータ」を参照してください。

## 仮想キュー テーブル

仮想キューのインタラクション処理に関連する情報を格納するテーブルは、G\_VIRTUAL\_QUEUEといます。このテーブルには、T-Serverがレポートするインタラクションと仮想キューの関連付けの履歴が格納されます(Universal Routing Server (URS)がこの情報をT-Serverに提供する場合)。

格納される仮想キュー データの詳細については、[61ページ](#)の「仮想キューとルーティング ポイントのモニタ」を参照してください。

---

## 構成テーブル

構成データを格納するテーブルには、以下の2つのサブグループがあります。

- オブジェクトテーブル
- オブジェクトリンク テーブル

以下のサブセクションで、それぞれのサブグループについて説明します。

## オブジェクト テーブル

オブジェクト テーブルの名前にはGC\_というプレフィックスが付きます。これらのテーブルには、ICONが追跡する構成オブジェクトについての現在(最新)の情報が格納されます。また、Configuration Databaseから削除された構成オブジェクトについての情報も、これらのテーブルに保持されます。

## オブジェクト リンク テーブル

オブジェクト リンク テーブルの名前にはGCX\_というプレフィックスが付きます。これらのテーブルには、Configuration Databaseの構成オブジェクト間の関連付け(リンク)についての情報が格納されます。構成オブジェクト間の関連付けの例には、エージェントへのスキルまたはログインの割り当て、およびエージェント グループへのエージェントの割り当てなどがあります。また、これらのテーブルには、エージェント グループからエージェントが削除されたという事実に関する情報など、終了した関連付けについての情報も保持されます。

---

## サービスおよび辞書テーブル

サービス テーブルと辞書テーブルは、ICONの内部処理や他のテーブルのフィールドを記述するために使用されます。

このグループには以下の5つのテーブルがあります。

- G\_DICTIONARY および G\_DICT\_TYPE—他のテーブルの特定の列挙型フィールドの辞書情報を格納 (STATE、STATUS、CAUSE フィールドなど)
- G\_DB\_PARAMETERS—データベース スキーマに関する一般情報を格納 (スキーマ バージョンなど)
- G\_HA\_CONTROL、G\_SYNC\_CONTROL、G\_PROV\_CONTROL—ICON が使用して内部トランザクション制御を実装
- G\_TIMECODE—他のテーブル(CREATED\_TCODE、DELETED\_TCODEなど)で参照されるタイムコード値を、月、曜日、日などの特定の時間値エンティティに拡張
- GSYS\_SYSPROCINFO、GSYS\_DNPREMOTELOCATION、G\_LOG\_MESSAGES、G\_LOG\_ATTRS、および G\_LOG\_GETIDRANGEREQ—ICON のシステム プロシージャが使用する内部テーブル





# 3

## Genesys Multimedia および Open Media との統合

Genesys Customer Interaction Management (CIM) プラットフォームの一部は、従来型ではない音声以外のメディア(電子メール、チャットなど)およびオープン(カスタム設計)メディア(ファックス、Webフォームなど)に関連するインタラクションを管理します。このような部分をGenesys Multimediaと呼びます(以前のMulti-Channel RoutingまたはMCR)。

この章では、マルチメディア インタラクションとオープン メディア インタラクションについてのデータを、IDBで使用可能にする方法を説明します。以下の項があります。

- [マルチメディア オブジェクト \(37 ページ\)](#)
- [マルチメディア インタラクションのデータ処理と格納 \(38 ページ\)](#)
- [オープン メディア インタラクションの処理 \(41 ページ\)](#)
- [isOnline チャット属性 \(41 ページ\)](#)
- [構成に関する推奨事項 \(43 ページ\)](#)

---

### マルチメディア オブジェクト

この項では、マルチメディア アクティビティについてのInteraction Concentrator データに関する用語と要素(オブジェクト)について説明します。

以下の項目について取り上げます。

- エンドポイント
- DN

## エンドポイント

ICON には、以下の論理エンドポイントについてのレポート データが格納されます。

- インタラクション キュー —Configuration Layer でスクリプト オブジェクト (インタラクション キュー タイプ) として構成
- インタラクション ワークビン —Configuration Layer でスクリプト オブジェクト (インタラクション ワークビン タイプ) として構成
- ルーティング ストラテジ —Configuration Layer でスクリプト オブジェクト (単純ルーティング タイプ) として構成
- エージェントの作業場所 —Configuration Layer で作業場所オブジェクトとして構成

ICONでは、スクリプト オブジェクトについての構成関連情報がGC\_SCRIPTテーブルに格納されます。

ICONでは、作業場所オブジェクトについての構成関連情報がGC\_PLACEテーブルに格納されます。

## DN

Interaction Serverは、MultimediaSwitchタイプの交換機構成オブジェクトを使用します。ICONでは、Interaction Server交換機で構成されたDNに関する情報が、他のタイプの交換機で構成されたDNに関する情報と同様の方法で格納されます。

ICONでは、DN構成オブジェクトについての構成関連情報がGC\_ENDPOINTテーブルに格納されます。

ICONでは、DNと作業場所の関連付けについての構成関連情報がGCX\_ENDPOINT\_PLACEテーブルに格納されます。

---

## マルチメディア インタラクションのデータ処理と格納

以下の項目について取り上げます。

- マルチメディア レポート プロトコル イベント (39 ページを参照)
- マルチメディア インタラクション (39 ページを参照)
- サポートされるシナリオ (40 ページを参照)

マルチメディア インタラクションのエージェント状態とログイン セッションに関する情報を収集する方法については、45ページの第4章を参照してください。

ICONがマルチメディア インタラクションの添付データを処理する方法については、58ページの「マルチメディア用添付データの処理」を参照してください。

ICONがマルチメディア データを格納するIDBテーブルの詳細については、ご使用のRDBMSに該当するものについて、『*Interaction Concentrator 7.6 Physical Data Model*』を参照してください。

## マルチメディア レポート プロトコル イベント

ICONがInteraction Serverに接続し、マルチメディア 電子メールおよびマルチメディア チャット インタラクション処理に関する通知を、マルチメディア レポート プロトコル イベントの形で受け取ります。

ICON は、以下に示すインタラクションのマルチメディア レポート プロトコル イベントを処理します。

- EventInteractionSubmitted
- EventPartyAdded
- EventProcessingStopped
- EventPartyRemoved
- EventPlacedInQueue
- EventPropertiesChanged
- EventTakenFromQueue
- EventAgentInvited
- EventPlacedInWorkbin
- EventRejected
- EventTakenFromWorkbin
- EventRevoked
- マルチメディア レポート カスタム メッセージ(仮想キュー関連TEventsのエンベロープ)

マルチメディア レポート プロトコル イベントの詳細については、『*Genesys 7 Events and Models Reference Manual*』を参照してください。

## マルチメディア インタラクション

ICON は、マルチメディア インタラクション処理に関する詳細を、IDB 内の音声レコードを格納するテーブルと同じテーブルに格納します。

- コア テーブル**
- **G\_CALL**— 電子メールまたはチャット メディア タイプの各インタラクションを1つのレコードとして表現。EventInteractionSubmittedを受信したときにICONがレコードを作成します。  
EventProcessingStoppedを受信したときに、ICONがレコードを更新(レコードを終了に変更)します。レコードがマルチメディア インタラクションの場合(つまり、Interaction Serverのイベント処理の結果、インタラクションレコードが作成された場合)、ICONがGSYS\_EXT\_INT1フィールドの値を1に設定します。
  - **G\_PARTY**— マルチメディア インタラクションに参加するパーティに関する情報を格納。エンドポイントとマルチメディア インタラクション間の関係が確立されたという通知をInteraction Serverから受け取ったときに、ICONが新規レコードを作成します。

---

**注:** マルチメディア インタラクションに参加する外部パーティのレコードは格納されません。

---

- **G\_IR**— 特定のシナリオのすべてのインタラクションに共通のデータに関する情報を格納。レコードがマルチメディア インタラクションの場合 (つまり、**Interaction Server** のイベント処理の結果、インタラクション レコードが作成された場合)、**ICON** が **GSYS\_EXT\_INT1** フィールドの値を 1 に設定します。

以下のいずれかの条件を満たすとき、マルチメディア インタラクションの新規レコードが作成されます。

- ◆ インタラクションが既存の親要素インタラクションと関連付けられていない。
- ◆ インタラクションが既存の親要素インタラクションと関連付けられているが、関連付けられた **G\_IR** 親要素レコードについての情報を入手できない。この場合、**Interaction Server** が **GSYS\_EXT\_VCH1** フィールドで親要素レコードについて提供する ID 情報を、**ICON** が格納します。
- **G\_ROUTE\_RESULT**— ストラテジについて **G\_PARTY** テーブルで作成されるレコードを参照し、インタラクションのルーティング結果に関する情報を格納

---

**注：** マルチメディア インタラクション データは **G\_IS\_LINK** テーブルには書き込まれません。

---

- |               |   |
|---------------|---|
| <b>履歴テーブル</b> | <ul style="list-style-type: none"> <li>• <b>G_CALL_HISTORY</b>、<b>G_PARTY_HISTORY</b>、および <b>G_IR_HISTORY</b>— 前にコア テーブルに格納されていたデータの間 (履歴) 状態を格納</li> </ul>   |
| <b>統計テーブル</b> | <ul style="list-style-type: none"> <li>• <b>G_CALL_STAT</b> および <b>G_PARTY_STAT</b>— それぞれ、マルチメディア インタラクションとパーティに関するメトリクスを格納。使用可能な事前に計算されたメトリクスについては、<a href="#">29 ページ</a>と <a href="#">30</a> の表 2 と 3 を参照してください。</li> </ul> |

## サポートされるシナリオ

**Interaction Concentrator** は、マルチメディア インタラクションについて以下のようなシナリオをサポートします。

- インタラクションの送信
- インタラクションの配信
- 転送
- 会議通話
- 自動受信確認
- 自動応答
- 処理なしの放棄 (チャットのみ)

## オープンメディアインタラクションの処理

ICONには、オープンメディアインタラクション処理についての詳細情報と、この処理に関連するエージェントアクティビティの詳細情報が格納されません。

Interaction Serverから受信する各インタラクションには、Configuration Serverで定義されたメディアタイプ名が含まれています。ICONがEventInteractionSubmittedなどのレポートイベントを処理するとき、インタラクションからメディアタイプ名を抽出し、この文字列を整数値にマップします。

ICONは、オープンメディアインタラクション処理に関する詳細を、IDB内の音声レコードとマルチメディアレコードを格納するテーブルと同じテーブルに格納します。

### コアテーブル

- G\_CALL
- GX\_SESSION\_ENDPOINT
- G\_AGENT\_STATE\_RC
- GS\_AGENT\_STAT
- G\_AGENT\_STATE\_HISTORY

上記の各コアテーブルで、オープンメディアインタラクションに対して、ICONが以下の値を設定します。

- GSYS\_EXT\_INT1 フィールド = 1000
- GSYS\_EXT\_VCH1 フィールド = メディアタイプ名を含む文字列値

さらに、ICONはG\_CALLテーブルのMEDIATYPEフィールド(= 1000)も設定します。

**注：** これらのテーブルの詳細については、[38 ページ](#)の「マルチメディアインタラクションのデータ処理と格納」を参照してください。

## isOnline チャット属性

isOnline属性は、チャットインタラクションをサポートするためにリリース7.6で導入されました。チャットセッションがアクティブになると、isOnline属性の値がonになり、Genesys Chat ServerがInteraction Serverに通知します。チャットセッションが終了すると(最後の構成員がチャットセッションから離脱すると)、isOnline属性の値がoffに変化し、Chat Serverがこの属性の変更をInteraction Serverに通知します。

isOnline属性のモニタと変更はChat Serverが行います。Chat Serverが切断したりシャットダウンした場合、Interaction ServerがInteraction Concentratorにチャットインタラクションの状況に関する通知を送信できません。[図2](#)に、isOnlineチャット属性の送信と状態の例を示します。

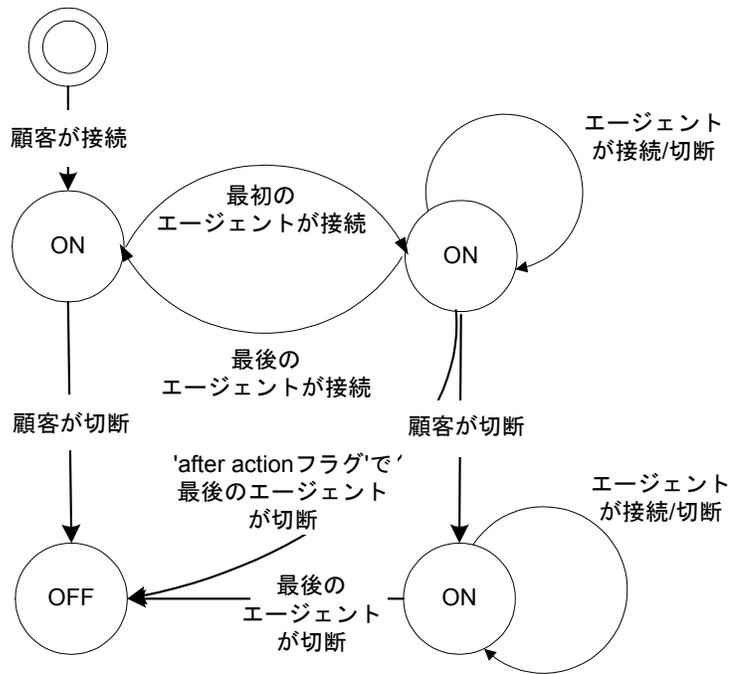


図2: チャットインタラクションの isOnline 属性

## isOnline 属性の処理

ICONはすべてのインタラクションレポートイベント (EventInteractionSubmitted、EventPropertiesChanged、および EventProcessingStopped)のisOnline属性の変化をモニタします。ICONは添付データの処理と同様のメカニズムを使用して、isOnline属性を格納します。つまり、G\_USERDATA\_HISTORYテーブルに新規レコードを作成し、isOnline属性の値を格納します。表4に、G\_USERDATA\_HISTORYテーブルのキーフィールドと有効値を示します。

表4: G\_USERDATA\_HISTORY テーブルのキー フィールド

フィールド名	値	注釈
KeyName	_attr_is_online	この値は変化しない。
ChangeType	_created	ICONがEventInteractionSubmittedを受け取ったときに格納される値。
	_updated	ICONがEventPropertiesChangedを受け取ったときに格納される値。
	_terminated	EventProcessingStoppedを受け取ったときに作成。

表 4: G\_USERDATA\_HISTORY テーブルのキー フィールド (つづき)

フィールド名	値	注釈
Type	Source_Attributes	内線、理由、添付データ(userdata)などのデータソース。値は以下のとおり。 1—userdata 2—理由 3—内線 4—属性(将来のために予約) 5—mcr_workbin
KeyID	9995	ハードコード値。
Value	0または1 (イベントから取得)	1—チャットセッションがアクティブ 0—チャットセッションが停止
Added	Timestamp	レコードが変更された時刻。

## 構成に関する推奨事項

レポートのためにマルチメディア インタラクション、エージェント状態、およびログインセッションデータをIDB格納するには、Genesys Configuration Layerで特定の構成を設定する必要があります。この項では、ICONアプリケーション オブジェクトで設定する必要がある構成について説明します。

## ICON アプリケーション

ICON がマルチメディア データを受信して IDB に格納するためには、適切な Interaction Server インスタンスとの ICON の接続を構成する必要があります。

**接続** ICONはInteraction Serverタイプのアプリケーション オブジェクトに直接接続することはできません。したがって、1つのInteraction Serverのみを使用する環境では、T-Serverタイプのアプリケーションを作成して、接続パラメータをInteraction Serverに設定します。ICONアプリケーション オブジェクトの[接続]タブで、このT-Serverアプリケーション オブジェクトに接続を追加します。

複数のInteraction Serverを使用する環境では、構築トポロジを決定する必要があります。具体的には、1つのICONインスタンスですべてまたはいくつかのInteraction Serverからのデータを処理するか、各Interaction Serverに専用のICONインスタンスを割り当てるかを決定します。この決定に基づいて、T-Serverタイプのアプリケーションをいくつか作成し、接続パラメータをInteraction Serverに設定します。Interaction Serverからのデータを格納するICONアプリケーション オブジェクトの[接続]タブに、必要なT-Serverアプリケーション オブジェクトを追加します。

## マルチメディア用の設定

他のICONアプリケーション オブジェクトの構成オプションには特別な要件はありません。マルチメディア オブジェクトとインタラクションについて、ICONが収集するデータのタイプは、ICONインスタンスに対して設定するrole構成オプションによって決定されます。構成オブジェクトの詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』を参照してください。

## オープン メディア用の構成

ICONがオープン メディア インタラクションについての情報をIDBに格納するためには、ICONアプリケーション オブジェクトのcallconcentratorセクションで、mcr-om-processing構成オプションを設定する必要があります。このオプションの設定の詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』を参照してください。



# 4

## 添付データの処理

この章では、ボイス コール、マルチメディア、およびオープン メディア インタラクションに添付されたユーザ データを Interaction Database (IDB) で使用方法について説明します。以下の項があります。

- [ボイス コールの添付データ処理 \(45 ページ\)](#)
- [カスタム添付データの処理 \(48 ページ\)](#)
- [構成に関する推奨事項 \(49 ページ\)](#)
- [添付データ仕様ファイル \(50 ページ\)](#)
- [マルチメディア用添付データの処理 \(58 ページ\)](#)
- [オープン メディア用の添付データ処理 \(60 ページ\)](#)

---

**注：** 添付データの処理と格納は多くのリソースと費用を必要とします。レポートとトラブルシューティングの要件を十分に検討し、Interaction Concentrator で収集する添付データの量を制限することをお奨めします。

---

---

## ボイス コールの添付データ処理

このセクションでは、Interaction Concentrator (ICON)がボイス コールに添付されたユーザ データを処理する方法を簡単に説明します。

### T-Server インタラクション

T-Serverからのデータを処理する際に、ICONがすべてのTEventsをチェックして、添付データに加えられた変更を確認します。特定のインタラクションの添付データが変更されると、ICONが変更内容を解析し、アプリケーション構成または添付データの指定に従って、データをIDBに格納します。

## 呼固有のデータ

呼固有のデータは、1つの呼だけに関連付けられた添付データです。指定されたタイムアウト後に呼が消去されると、このデータがIDBに格納されます。call-deletion-timeout構成オプションを設定して、タイムアウト間隔を指定する詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』の構成についての章を参照してください。

## 履歴データ

履歴データは添付データと関連付けられます。履歴データは以下のような関連付けを持ちます。

- パーティ —EventAttachedDataChanged でサードパーティ DN がパーティを表す場合
- エンドポイント—EventAttachedDataChangedでサードパーティDNが指定されている場合
- エージェント—データが変更されたときに、エージェントが特定のデバイスに関連付けられている場合

格納される変更タイプには、allの履歴タイプを持つレコードの変更タイプが反映されます。履歴タイプの詳細については、[53ページ](#)の「履歴タイプ」を参照してください。

[表5](#)に、添付データ変更タイプの値の定義を示します。

**表 5: 添付データ変更タイプ**

変更タイプ	対応する数値	説明
created	1	指定されたキーの呼が作成された。
added	2	既存の添付データにキーが追加された。
updated	3	既存のキーが更新された。
deleted	4	既存のキーが削除された。
terminated	5	既存のキーの呼が終了した。

## 音声に添付されたデータのデータベース スキーマ拡張

IDBにはUserData収集のために、以下の2タイプのテーブルがあります。

- いわゆるフラットテーブル。特定のキー名に対応するデータがテーブルの特定のフィールドに格納されます。
- 汎用または履歴テーブル。添付データの値それぞれが、対応するコンテキストの個別の行に格納されます。

表6に、ボイス コールに添付されたユーザ データを格納するためのIDBテーブルを示します。

**注：** 添付データ格納テーブルでは、それぞれの呼に関するユーザ データが個別のレコードに格納されます(1つのレコードに1つの呼)。

表 6: 音声の添付データ格納テーブル

テーブル名	データのタイプ	説明
G_CALL_USERDATA	呼	呼の存続期間全体にわたって収集される事前定義されたUserData用。
G_CALL_USERDATA_CUST	呼	呼の存続期間全体にわたって収集されるカスタムのUserData用。
G_CALL_USERDATA_CUST1	呼	呼の存続期間全体にわたって収集されるカスタムのUserData用。
G_CALL_USERDATA_CUST2	呼	呼の存続期間全体にわたって収集されるカスタムのUserData用。
G_USERDATA_HISTORY	履歴	UserDataに加えられた履歴の変更を格納する。
G_SECURE_USERDATA_履歴	履歴	UserDataに加えられた履歴の変更を格納する。このテーブルの許可は特定のサイトで設定すること。

## 添付データのセキュリティ

Interaction Concentratorでは、添付データを格納するテーブルをいくつか用意することで、機密性の高いユーザ データを保護します。DBA (Database Administrator: データベース管理者)がユーザ権限を割り当てて、保護されたユーザ データ履歴テーブルへのアクセスを制限できます。同様に、DBAがフラットテーブルへのアクセスも制限できます。

## マルチサイト環境と分散環境

マルチサイト環境や地理的に分散した環境では、2つのT-Serverに添付データが配信されると、それぞれのT-Serverが添付データを独自のIDBに格納します。その結果、サイト間のIDBで添付データが複製されることになります。

## カスタム添付データの処理

カスタム ストアド プロシージャ または カスタム ディスパッチャ を作成して、ボイス コール に添付された ユーザ データ を処理し、添付データを IDB のカスタム テーブル に格納することができます。呼が終了すると、ICON がカスタム ディスパッチャ を呼び出します。

**データ型** ICON では以下の 2 つの型の添付データの値を処理できます。

- 文字列
- 整数

**キー グループ** 同じグループで指定されるキー名の値は、カスタム ディスパッチャ ストアド プロシージャ に対する同じ呼び出しによって渡されます。添付データ構成ファイル内でキーのグループを指定できます。各グループ内で、最大17の文字列の KVP (Key-Value Pair : キーと値のペア) および17の整数値の KVP を指定できます。ICON が処理するキー グループの最大数を設定できます(『*Interaction Concentrator 7.6 Deployment Guide*』の `gud-cust-disp-groups` 構成オプションの説明を参照)。

カスタム添付データ処理のためのXML仕様の例については、[57ページ](#)の「カスタム添付 データの仕様のサンプル」を参照してください。

## カスタム ディスパッチャ

IDB 初期化スクリプトによって、以下の 2 つのカスタム ディスパッチャ ストアド プロシージャ が作成されます。

- `gudCustDisp1`
- `gudCustDisp2`

ICON アプリケーション オブジェクトの `gud-cust-disp` で、ICON が呼び出すカスタム ディスパッチャ を指定します(『*Interaction Concentrator 7.6 Deployment Guide*』の構成オプションの章のオプションの説明を参照)。

ICON の実行中に、1 つのディスパッチャ から別のディスパッチャ に切り替えることができます。これによって、添付データの処理を中断せずに、カスタム ディスパッチャ 構成に変更を加えることができます。

デフォルトのカスタム ディスパッチャ は何も実行しません。スクリプトを変更してカスタム ディスパッチャ を作成し、必要な添付データを格納します。また、必要なカスタム テーブルを IDB に作成するスクリプトも記述する必要があります。

**サンプルスクリプト** IDB 初期化スクリプトに加えて、Interaction Concentrator インストール パッケージには以下のサンプルスクリプトも用意されています。

- `sample_gcc_<db_type>_custdisp_schema.sql`— カスタム添付データ格納テーブル (`G_SAMPLE_CUST_ADATA`) の作成方法を示したサンプルスクリプト。

- `sample_gcc_<db_type>_custdisp_api.sql`—`gudCustDisp1` ストアドプロシージャの変更方法を示したサンプル スクリプト。変更されたストアドプロシージャには `G_SAMPLE_CUST_ADATA` テーブルの引数が格納されます。

**注：** サンプル スクリプトは、カスタム ディスパッチャ ストアド プロシージャの変更方法を示すために用意されています。インストール時にサンプル スクリプトを実行することは避けてください。

変更した `gudCustDisp1` または `gudCustDisp2` ストアドプロシージャの構文と演算を十分確認してください。カスタム ディスパッチャ ストアドプロシージャによって何らかのエラーや RDBMS 違反が発生すると、ICON によるボイス コールとマルチメディア インタラクションの他の添付データの処理にも影響する可能性があります。

## 構成に関する推奨事項

以下の 2 つのプロセスで、添付データを IDB に格納するように Interaction Concentrator を構成します。

1. 添付データ キー構成ファイルを指定します。これは、レポート イベント属性の KVP を IDB のテーブルとフィールドにマップします。詳細については、[50 ページ](#)の「添付データ仕様ファイル」を参照してください。
2. Genesys Configuration Layer で、添付データの構成設定を指定します。

この項では、ICON アプリケーション オブジェクトで設定可能な構成について説明します。

### ICON の役割

添付データを格納する ICON インスタンスそれぞれについて、ICON アプリケーション オブジェクトの [オプション] タブで、`role` オプションの値のリストに `gud` が含まれていることを確認します。コンタクトセンター全体に1つの ICON インスタンスが構築されている場合は、デフォルト値 (`all`) のままにします。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』の `role` 構成オプションの説明を参照してください。

### 添付データ仕様ファイル

以下の ICON 構成オプションで、ICON に対して異なる添付データ仕様ファイルを指定できます。

- `adata-spec-name`

### 添付データ構成オプション

以下の ICON 構成オプションで、ICON が格納する添付データと格納方法を指定できます。

- `adata-default-storage`
- `adata-extensions-history`
- `adata-reasons-history`
- `adata-userdata-history`

『*Interaction Concentrator 7.6 Deployment Guide*』を参照して、添付データ構成オプションの説明と値について確認してください。ご使用の環境に適切な値を選

択し、ICONアプリケーション オブジェクトの[オプション]タブで関連する構成を変更する必要があります。

#### カスタム ディス パッチャ構成 オプション

以下の ICON 構成オプションで、カスタム ディスパッチャが添付データを処理する方法を指定できます。

- gud-cust-disp
- gud-cust-disp-groups

『*Interaction Concentrator 7.6 Deployment Guide*』を参照して、カスタム ディスパッチャ構成オプションの説明と値について確認してください。ご使用の環境に適切な値を選択し、ICONアプリケーション オブジェクトの[オプション]タブで関連する構成を変更する必要があります。

---

## 添付データ仕様ファイル

この項では、T-Server が TEvents を使用して送信する添付データの KVP を処理するための XML スキーマ定義について取り上げます。ICON を使用して添付データを IDB に格納する場合は、この項の定義に基づいて、ICON が使用する添付データ仕様を作成してください。

添付データ仕様はXMLファイルで、Interaction Concentratorアプリケーションをインストールする際に指定したインストール ディレクトリに格納されています(『*Interaction Concentrator 7.6 Deployment Guide*』のICONのインストールと構成についての章を参照)。

添付データ仕様のサンプルについては、[56ページ](#)の「基本添付データ仕様のサンプル」を参照してください。

## 解析の制限

ICONのXML解析には以下の制限があります。

- 仕様に不明な属性がある場合、無視されます。XML 仕様を解析するとき、ICON は抜けている属性だけをチェックします。
- ICON の XML 解析は名前空間をサポートしません。
- 重複したキーは無視されます。最初に出現するキー名のみを使用して、データベース テーブルの指定のフィールドが更新されます ([56ページ](#)の注と例を参照)。

## スキーマ定義

添付データ仕様のXMLスキーマ定義を以下に示します。

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema targetNamespace="http://www.genesyslab.com/standards/icon/ed1"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:icon="http://www.genesyslab.com/standards/icon/ed1" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
<xsd:annotation>
  <xsd:documentation>Attached data configuration specification</xsd:documentation>
</xsd:annotation>

<xsd:simpleType name="AdataHistoryType">
<xsd:restriction base="xsd:string">
  <xsd:enumeration value="none" />
<xsd:enumeration value="first" />
<xsd:enumeration value="last" />
<xsd:enumeration value="all" />
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="AdataCallHistoryType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="first" />
<xsd:enumeration value="last" />
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="AdataCallFieldType">
<xsd:restriction base="xsd:string">
  <xsd:enumeration value="customer-segment" />
<xsd:enumeration value="service-type" />
  <xsd:enumeration value="service-subtype" />
  <xsd:enumeration value="busines-result" />
  <xsd:enumeration value="customer-id" />
  <xsd:enumeration value="transaction-id" />
  <xsd:enumeration value="cause-id" />
  <xsd:enumeration value="account-id" />
  <xsd:enumeration value="destination-id" />
  <xsd:enumeration value="target-id" />
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="AdataCallCustFieldType">
<xsd:restriction base="xsd:string">
  <xsd:enumeration value="cust-data-1" />
  <xsd:enumeration value="cust-data-2" />
  <xsd:enumeration value="cust-data-3" />
  <xsd:enumeration value="cust-data-4" />
  <xsd:enumeration value="cust-data-5" />
  <xsd:enumeration value="cust-data-6" />
</xsd:restriction>
</xsd:simpleType>
```

```
<xsd:enumeration value="cust-data-7" />
<xsd:enumeration value="cust-data-8" />
<xsd:enumeration value="cust-data-8" />
<xsd:enumeration value="cust-data-10" />
<xsd:enumeration value="cust-data-11" />
<xsd:enumeration value="cust-data-12" />
<xsd:enumeration value="cust-data-13" />
<xsd:enumeration value="cust-data-14" />
<xsd:enumeration value="cust-data-15" />
<xsd:enumeration value="cust-data-16" />
<xsd:enumeration value="cust-data-17" />
<xsd:enumeration value="cust-data-18" />
<xsd:enumeration value="cust-data-19" />
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="AdataHistoryRecord">
  <xsd:attribute name="name" type="xsd:string" />
  <xsd:attribute name="source" type="icon:AdataSource" />
  <xsd:attribute name="history" type="icon:AdataHistoryType" />
</xsd:complexType>
<xsd:complexType name="AdataCallRecord">
  <xsd:attribute name="name" type="xsd:string" />
  <xsd:attribute name="source" type="icon:AdataSource" />
  <xsd:attribute name="history" type="icon:AdataCallHistoryType" />
  <xsd:attribute name="field" type="icon:AdataCallFieldType" />
</xsd:complexType>
<xsd:complexType name="AdataCallCustRecord">
  <xsd:attribute name="name" type="xsd:string" />
  <xsd:attribute name="source" type="icon:AdataSource" />
  <xsd:attribute name="history" type="icon:AdataCallHistoryType" />
  <xsd:attribute name="field" type="icon:AdataCallCustFieldType" />
</xsd:complexType>
<xsd:element name="adata_spec">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="public" type="icon:AdataHistoryRecord" />
      <xsd:element name="secure" type="icon:AdataHistoryRecord" />
      <xsd:element name="call" type="icon:AdataCallRecord" />
      <xsd:element name="call-cust" type="icon:AdataCallCustRecord" />
      <xsd:element name="call-cust1" type="icon:AdataCallCustRecord" />
      <xsd:element name="call-cust2" type="icon:AdataCallCustRecord" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:schema>
```

## 属性値

この項では、XMLスキーマ定義で使用される属性について説明します。

### 履歴タイプ

履歴タイプとして、以下の値を使用できます。

none	IDBに所定のキーの値が記録されません。
first	IDBに所定のキーの最初の値のみが記録されます。
last	IDBに所定のキーの最後の値のみが記録されます。
all	IDBに所定のキーの値の変更すべてが記録されます。この値は、履歴テーブルに格納するために構成されたキーのみに適用されます。

### 格納タイプ

表7に、各属性が格納されるデータベース テーブルを示します。

表7: 添付データの格納タイプ

属性名	IDB テーブル名
public	G_USERDATA_HISTORY
secure	G_SECURE_USERDATA_HISTORY
call	G_CALL_USERDATA
call-cust	G_CALL_USERDATA_CUST
call-cust1	G_CALL_USERDATA_CUST1
call-cust2	G_CALL_USERDATA_CUST2

### データ ソース タイプ

表8に、各属性の取得元のTEvent属性を示します。

表8: 添付データ属性のソース タイプ

属性名	TEvent 属性名
reasons	AttributeReasons
extensions	AttributeExtensions
userdata	AttributeUserData

### 事前定義のフィールド

表9に、各属性が格納される事前定義されたIDBフィールドを示します。

表9: 事前定義の添付データ フィールド

属性名	G_CALL_USERDATA フィールド名
customer-segment	G_CUSTOMER_SEGMENT
service-type	G_SERVICE_TYPE

表 9: 事前定義の添付データ フィールド (つづき)

属性名	G_CALL_USERDATA フィールド名
service-subtype	G_SERVICE_SUBTYPE
business-result	G_BUSINESS_RESULT
customer-id	CUSTOMER_ID
transaction-id	TRANSACTION_ID
cause-id	CAUSE_ID
account-id	ACCOUNT_ID
destination-id	DESTINATION_ID
target-id	TARGET_ID

**カスタム フィールド** 表10に、添付データ仕様で使われるカスタム属性に対して、ICONが作成するIDBフィールドを示します。

表 10:事前定義の添付データ フィールド

属性名	G_CALL_USERDATA_CUST* フィールド名
cust-data-1	CUST_DATA_1
cust-data-2	CUST_DATA_2
cust-data-3	CUST_DATA_3
cust-data-4	CUST_DATA_4
cust-data-5	CUST_DATA_5
cust-data-6	CUST_DATA_6
cust-data-7	CUST_DATA_7
cust-data-8	CUST_DATA_8
cust-data-9	CUST_DATA_9
cust-data-10	CUST_DATA_10
cust-data-11	CUST_DATA_11
cust-data-12	CUST_DATA_12
cust-data-13	CUST_DATA_13
cust-data-14	CUST_DATA_14

表 10:事前定義の添付データ フィールド(つづき)

属性名	G_CALL_USERDATA_CUST* フィールド名
cust-data-15	CUST_DATA_15
cust-data-16	CUST_DATA_16
cust-data-17	CUST_DATA_17
cust-data-18	CUST_DATA_18
cust-data-19	CUST_DATA_19

**事前構成されたキー**

以下の添付データ キーに対して、ICONでは履歴タイプnoneが事前に設定されています。

- RTargetRuleSelected
- RTargetObjectSelected
- RTargetTypeSelected
- RTargetAgentSelected
- RTargetPlaceSelected
- RStrategyName
- RRequestedSkillCombination
- RTenant

デフォルトでは、これらのキーの値はIDB履歴テーブルに格納されません。このデータの一部またはすべてを格納する必要がある場合は、添付データ仕様でそれぞれのキーを明示的に定義してください。

---

**注：** RTargetRuleSelected キーを格納する場合は、値が個別のテーブル G\_ROUTE\_RESULT に格納されます。

---

## 基本添付データ仕様のサンプル

以下に示すのは、ボイス コールの基本的な adata\_spec.xml ファイルの例です。

```
<?xml version="1.0" encoding="utf-8" ?>
<adata_spec>
<public>
  <key name = "u_key1" source="userdata" history ="all"/>
</public>
<secure>
  <key name = "u_key2" source="userdata" history ="all"/>
</secure>
<call>
  <key name = "customer-segment" source="userdata" history ="first" field="customer-segment"/>
  <key name = "svc_class_cd" source="userdata" history ="first" field="service-type"/>
  <key name = "CCTP_CALLTYPE" source="userdata" history ="first" field="service-subtype"/>
  <key name = "cid" source="userdata" history ="first" field="customer-id"/>
  <key name = "transact_tn_final" source="userdata" history ="first" field="transaction-id"/>
</call>
<call-cust>
  <key name = "customer-segment0" source="userdata" history ="first" field="cust-data-2"/>
  <key name = "STATE" source="userdata" history ="last" field="cust-data-1"/>
</call-cust>
<call-cust1>
  <key name = "customer-segment1" source="userdata" history="first" field="cust-data-1"/>
  <key name = "PegTD" source="userdata" history ="last" field="cust-data-2"/>
</call-cust1>
<call-cust2>
  <key name = "customer-segment2" source="userdata" history="first" field="cust-data-4"/>
  <key name = "vrapp_ctl_lang" source="userdata" history ="last" field="cust-data-3"/>
</call-cust2>
</adata_spec>
```

**重要！ 注：** ICON では重複したキーは無視されます。最初に出現したキー名だけを使用して、指定したデータベース テーブルが更新されます。

以下の例では、G\_CALL\_USERDATA\_CUSTテーブルのcust\_data\_2フィールドだけに、key name = customer-segmentに対応する値が格納されます (userdataに存在する場合)。G\_CALL\_USERDATA\_CUST1のcust\_data\_2フィールドは更新されません。

**例**

```
<call-cust>
<key name = "customer-segment" source="userdata" history
="first" field="cust-data-2"/>
<key name = "STATE" source="userdata" history ="last"
field="cust-data-1"/>
</call-cust>
```

```
<call-cust1>
  <key name = "customer-segment" source="userdata" history
    ="first" field="cust-data-1"/>
  <key name = "PegTD" source="userdata" history ="last"
    field="cust-data-2"/>
</call-cust1>
```

## カスタム添付データの仕様のサンプル

以下に示すのは、カスタム添付データを処理するために変更した  
adata\_spec.xml ファイルの例です。

```
<?xml version="1.0" encoding="utf-8" ?>
<adata_spec>
  <cust-disp-group-1>
    <key name = "name1_1" source="userdata" history ="first" field="cust-int-1"></key>
    <key name = "name2_1" source="userdata" history ="last" field="cust-int-2"></key>
    ...
    <key name = "name17_1" source="userdata" history ="last" field="cust-int-17"></key>
    <key name = "name18_1" source="userdata" history ="last" field="cust-str-1"></key>
    <key name = "name19-1" source="userdata" history ="last" field="cust-str-2"></key>
    ...
    <key name = "name34_1" source="userdata" history ="last" field="cust-str-17"></key>
  </cust-disp-group-1>
  <cust-disp-group-2>
    <key name = "name1_2" source="userdata" history ="first" field="cust-int-1"></key>
    <key name = "name2_2" source="userdata" history ="last" field="cust-int-2"></key>
    ...
    <key name = "name17_2" source="userdata" history ="last" field="cust-int-17"></key>
    <key name = "name18_2" source="userdata" history ="last" field="cust-str-1"></key>
    <key name = "name19_2" source="userdata" history ="last" field="cust-str-2"></key>
    ...
    <key name = "name34_2" source="userdata" history ="last" field="cust-str-17"></key>
  </cust-disp-group-2>
  ...
  <cust-disp-group-16>
    <key name = "name1_3" source="userdata" history ="first" field="cust-int-1"></key>
    <key name = "name2_3" source="userdata" history ="last" field="cust-int-2"></key>
    ...
    <key name = "name17_3" source="userdata" history ="last" field="cust-int-17"></key>
    <key name = "name18_3" source="userdata" history ="last" field="cust-str-1"></key>
    <key name = "name19_3" source="userdata" history ="last" field="cust-str-2"></key>
    ...
    <key name = "name34_3" source="userdata" history ="last" field="cust-str-17"></key>
  </cust-disp-group-16>
</adata_spec>
```

## マルチメディア用添付データの処理

この項では、マルチメディア インタラクション(電子メールとチャット)に添付されたユーザ データをICONが処理する方法について簡単に説明します。

### Interaction Server のインタラクション

Interaction Serverからのデータを処理する際に、ICONがすべてのマルチメディア レポート プロトコル イベントをチェックして、添付データに加えられた変更を確認します。特定のインタラクションの添付データが変更されると、ICONが変更内容を解析し、アプリケーション構成と添付データ仕様に従って、データをIDBに格納します。

### マルチメディア インタラクション固有のデータ

マルチメディア インタラクション固有のデータは、マルチメディア インタラクションだけに関連付けられた添付データです。ICONでは、このデータがIDBの個別のマルチメディア添付データ テーブルの事前定義のフィールドに格納されます。

**注：** デフォルトでは、Interaction Server によって自動的にキーは追加されません。このキーは、Interaction Concentrator がサポートするすべてのマルチメディア添付データについて通知するために必要です。場合によっては、Interaction Server が必要なキー(推奨応答の名前など)のデータを添付するように、ルーティング ストラテジを変更する必要があります。

### マルチメディア添付データのデータベース スキーマ拡張

マルチメディア固有のユーザ データを収集するために、IDB には以下の2つのテーブルがあります。

- GM\_L\_USERDATA—添付データ キーの値を格納(推奨応答、自動応答、受信確認、顧客 ID、処理停止の理由)。Interaction Server がインタラクションの完了を通知すると、ICON がこのテーブルに情報を書き込みます。
  - ICON は、Interaction Server から最初にインタラクションについての通知を受け取ったときに、該当するキーが持つ値から応答と受信確認の名前を収集します。
  - ICON は、Interaction Server がインタラクションの終了を通知したときに、顧客 ID と理由の情報を該当するキーの最後の値から収集します。
- GM\_F\_USERDATA—電子メールおよびチャット インタラクション(該当する場合)について、以下の情報を格納。
  - 送信者 (“From” の名前と電子メール アドレス)
  - コールバック
  - 件名

- ◆ タイプとサブタイプ
- ◆ 発信元ソース (Web フォームまたは電子メール)
- ◆ E-mail Server によりダウンロードされた時刻

Interaction Server がインタラクションに関するレポート イベントを最初に ICON に送信したときに、ICON が情報をこのテーブルに書き込みます。したがって、ICON が格納する値は、該当するキーについて提示される最初の値です。

## マルチメディア用の添付データ仕様ファイル

以下に示すのは、マルチメディア インタラクション用に変更した adata\_spec.xml ファイルの例です。

```
<?xml version="1.0" encoding="utf-8" ?>
<adata_spec>
<mcr-f-data>

    <key name = " FromPersonal" source="userdata" field="from-name"/>
    <key name = "CalledBack" source="userdata" field="called-back"/>
    <key name = "Subject" source="userdata" field="email-subject"/>
    <key name = "Origination_Source" source="userdata" field="origination-source"/>
    <key name = " FromAddress" source="userdata" field="from-address"/>
    <key name = "attr_itx_subtype" source="attributes" field="sub-type"/>
    <key name = "attr_itx_received_at" source="attributes" field="received-at"/>

</ mcr-f-data>
<mcr-l-data>

    <key name = "SuggestedResponseID" source="userdata" history ="first" field="suggested-
response"/>
    <key name = "AutoResponseID" source="userdata" history ="first" field="auto-response"/>
    <key name = " AutoAcknowledgementID" source="userdata" history ="first" field="auto-ask"/>
    <key name = "attr_reason_system_name" source="attributes" history ="last" field="stop-
reason"/>
    <key name = "ContactId" source=" userdata" history ="last" field="Multimedia-contact-id"/>

</ mcr-l-data>
</adata_spec>
```

---

## オープンメディア用の添付データ処理

この項では、オープンメディアインタラクション(ファックス、Webフォームなど)に添付されたユーザデータをICONが処理する方法について、簡単に説明します。

### オープンメディアインタラクション固有のデータ

オープンメディアインタラクション固有のデータは、オープンメディアインタラクションだけに関連付けられた添付データです。ICONでは、オープンメディア添付データに関する情報が、GM\_L\_USERDATAおよびGM\_F\_USERDATAテーブルに格納されます(これらのテーブルの詳細については、[58ページ](#)を参照)。これらのIDBテーブルやその他のIDBテーブルについては、ご使用のRDBMS (Relational Database Management System)のタイプに該当するものについて、『*Interaction Concentrator 7.6 Physical Data Model*』を参照してください。



# 5

## 仮想キューとルーティング ポイントのモニタ

この章では、Interaction Concentratorがどのように仮想キューとルーティング ポイントをモニタし、ルーティング情報をInteraction Database (IDB)に格納するかを説明します。仮想キュー データの格納のためにICONでサポートされる2つのモード、およびGenesys Configuration Layerの仮想キュー機能に関連した構成設定の推奨事項について取り上げます。

この章には以下の項があります。

- [仮想キューのルート結果のモニタ \(61 ページ\)](#)
- [ルーティング ポイントのルート結果のモニタ \(64 ページ\)](#)
- [構成に関する推奨事項 \(66 ページ\)](#)

---

### 仮想キューのルート結果のモニタ

Interaction Concentrator (ICON)は以下の処理を実行できます。

- コンタクト センターで構成され、ルーティングのために使用される仮想キュー オブジェクトのモニタ。Universal Routing Server (URS) が T-Server を介して、関連するデータを Interaction Concentrator に提供します。
- 仮想キューとキューに格納されるインタラクション間の関連付けを、IDB の特別なテーブルに個別のレコードとして格納。

この項では、仮想キューに関連するTEventsをInteraction Concentratorが処理する方法、およびInteraction Concentratorが格納する仮想キュー データの種類と格納方法について説明します。

---

**注：** IDB で使用可能な仮想キュー データの詳細については、ご使用の RDBMS に該当するものについて、『*Interaction Concentrator 7.6 Physical Data Model*』を参照してください。

---

## 格納モード

Interaction Concentrator では、仮想キュー データの格納のために、以下の 2 つのモードがサポートされます。

- ワンステップ格納
- ツーステップ格納

### ワンステップ格納

デフォルトのワンステップ格納モードでは、Interaction Concentratorがすべてのデータを強制的に1つのデータベース トランザクションに結合します。このモード以外の場合、これらのデータはレコード作成とレコード更新の個別のステップで格納されます。このモードでは、仮想キューとインタラクション間の関連付けが終了したときに、ICONがG\_VIRTUAL\_QUEUE IDBテーブルにレコードを作成します。つまり、ICONがEventDivertedまたはEventAbandonedのTEventを受け取った後です。これが推奨の格納モードです。このモードでは、IDBへの挿入回数が最小限になるため、パフォーマンスが向上します。ただし、仮想キューからエージェントのDNへインタラクションを迅速に送信する必要があります。

### ツーステップ格納

ツーステップ格納モードでは、EventQueuedを受け取った後で、まず、ICONがレコードを作成し、EventDivertedまたはEventAbandonedを受け取った後でICONがレコードを更新します。この格納モードは、インタラクションが仮想キューに長時間格納される環境に最適です。

## データ処理ステップ

仮想キュー データの処理を詳しく説明するために、この項ではツーステップ格納モードについて説明します。

### ステップ1— レコード作成

Interaction Concentrator のモニタ対象として設定された特定の仮想キューに関連する EventQueued TEvent を受信すると、G\_VIRTUAL\_QUEUE IDB テーブルに新しい行が挿入されます。格納されるデータはTEventおよび Configuration Database から取得され、以下の項目に関する情報が含まれています。

- T-ServerがレポートするCallUUIDの形式のインタラクション。これによって、後でレポートのために元のインタラクションを識別します。
- インタラクションが経由された交換機。これは、Configuration Server が対応する交換機オブジェクトに割り当てるデータベース ID の形式です(使用可能な場合)。

- インタクションが格納される仮想キュー。形式は、EventQueued の ThisDN 属性によってレポートされる番号、および Configuration Server が仮想キューに対応するDNオブジェクトに割り当てるデータベースIDです。さらに、Interaction Concentrator は以下の項目も格納します。
- 関連付けの状況。値は 8 で、キュー内待機を意味します。
- 仮想キューの状態変化の理由。値は 1 で、正常を意味します。
- 関連付けが作成された時刻。値は EventQueued が到着した時刻です。

## ステップ2—レコード更新

EventDivertedまたはEventAbandoned TEventが到着し、これが、すでにレコードが作成されている仮想キューとインタクションに関連している場合は、Interaction ConcentratorがG\_VIRTUAL\_QUEUE IDBテーブルのこのレコードを更新します。

ICONがG\_VIRTUAL\_QUEUE IDBテーブルの以下のデータを更新します。

- STATUS フィールドに格納される関連付けの状況。状況が以下のいずれかに変化します。
  - ◆ 13—転送済み。CallState=0 の EventDiverted が到着した場合は、この仮想キューから呼が転送されました。
  - ◆ 1—接続解除。CallState=22 の EventDiverted が到着した場合は、呼が別の仮想キューから転送されました。このキューの場合は、EventAbandoned が到着します。
- 仮想キューの状態の変化の原因は、CAUSE フィールドに格納されます。EventDiverted の場合は、以下のいずれかが原因となります。
  - ◆ 1—正常。ストラテジのターゲット選択オブジェクトで定義されたターゲット宛先へ、インタクションがルーティングされました。
  - ◆ 3—滞留。インタクションが仮想キューに滞留した後で、レコードが処理されました。
  - ◆ 101—並行仮想キューでインタクションがターゲット宛先にルーティングされました (注を参照)。
  - ◆ 102—URS によって、インタクションが URS 7.6 構成オプションで定義されたデフォルト宛先にルーティングされました (注を参照)。
  - ◆ 103—交換機によって、インタクションがデフォルト宛先にルーティングされました (注を参照)。
  - ◆ 104—URS ストラテジの ClearTarget 機能によって、インタクションが仮想キューから消去されました (注を参照)。
  - ◆ 105—その他。URS がレポートするその他の(分類されていない)原因を意味します (注を参照)。
  - ◆ 133—Interaction Server で設定されたインタクションのルーティングのタイムアウトがすぎました (注を参照)。
  - ◆ 134—Interaction Serverによって、インタクションがストラテジから削除され(取り出され)ました (注を参照)。

---

**注：** この値を CAUSE フィールドに格納するには、ICON 7.6 アプリケーションオブジェクトで、extended-route-result 構成オプションを 1 に設定する必要があります。Universal Routing Server (URS) リリース 7.6 および Interaction Server リリース 7.6.000.18 (以上) も必要です。

---

- EventAbandonedの場合は、CAUSEフィールドに格納される仮想キューの状態の変化の原因が、以下のように変化します。
  - ◆ 2—放棄

ICON は、レコードに以下のようなデータも追加します。

- 転送または放棄されたインタラクションの ID。形式は T-Server がレポートする CallUUID。これは、配信または放棄の瞬間にインタラクションが物理デバイスで持っていたものです (入手可能な場合)。後でレポートのために、この値によって配信または放棄されたインタラクションを識別します。
- インタラクションの配信先の交換機またはインタラクションが放棄された交換機。形式は、Configuration Server が対応する交換機オブジェクトに割り当てたデータベース ID です (入手可能な場合)。
- インタラクションの配信先の DN。形式は、EventDiverted の ThirdPartyDN 属性でレポートされる番号です (インタラクションがこの仮想キューから配信され、DN に関する情報が入手可能な場合)。
- 関連付けが終了した時刻。EventQueued または EventAbandoned が到着した時刻と同じです。
  - ◆ 元のインタラクションについての情報、インタラクションが経由された交換機、仮想キュー、および関連付けの開始時刻は、更新の際に変更されません。

---

## ルーティングポイントのルート結果のモニタ

そのように設定されている場合は、URS 7.6 で、ルーティングポイントから配信されたインタラクションとルーティングキューから配信されたインタラクションのルーティング結果を区別できます。Interaction Concentrator 7.6 が、このように拡張されたルーティング結果を T-Server 7.6 を介して URS から受け取り、IDB テーブル ROUTE\_RESULT の RESULT フィールドに格納できます (表 11 を参照)。

---

**注：** この機能をサポートするためには、66 ページに記載されたように、URS と ICON の構成オプションを設定する必要があります。

---

表 11: G\_ROUTE\_RESULT に格納される値の要約

レポート イベント	RESULT フィールドの値	格納される結果の説明
<b>extended-route-result = 0</b> の場合(ICONリリース7.5の機能)		
EventRouteUsed	1 (正常、ROUTE_RESULT_SUCCESS)	呼/インタラクションがルーティングされた。
EventAbandonedまたはEventPartyRemoved	2 (放棄、ROUTE_RESULT_FAIL)	呼が放棄された、またはインタラクションが削除された。
<b>extended-route-result = 1</b> の場合(ICONリリース7.6の機能)		
EventRouteUsed	1 (正常、ROUTE_RESULT_SUCCESS)	URS ストラテジによって呼/インタラクションがルーティングされた。
	102 (タイムアウト)	タイムアウトした後で呼がデフォルト宛先にルーティングされた、またはストラテジ内でTRoute[DN] 関数が呼び出されて呼が指定のDNへルーティングされた。
	103 (交換機によるルーティング)	交換機によって呼がデフォルトの場所へルーティングされた。
EventAbandoned	2 (放棄、ROUTE_RESULT_FAIL)	呼が放棄された。
EventPartyRemoved	1 (正常、ROUTE_RESULT_SUCCESS)	インタラクションがルーティングされた。
	2 (放棄、ROUTE_RESULT_FAIL)	インタラクションが停止された。
	134	Interaction Serverがインタラクションを取り出した。
EventPartyRemoved (つづき)	133	Interaction Serverで定義されたルーティングタイムアウトがすぎた。
	105	URSがインタラクションをルーティングしようとしているときに、Interaction ServerとURS間の接続が失われた。

**注：** extended-route-result 構成オプションの詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』を参照してください。IDB テーブルについては、ご使用の RDBMS (Relational Database Management System) のタイプに該当するものについて、『*Interaction Concentrator 7.6 Physical Data Model*』を参照してください。

## 信頼性フラグ

ICONは、G\_ROUTE\_RESULTテーブルのGSYS\_EXT\_INT1フィールドに格納される信頼性フラグを使用します。このフラグは、G\_ROUTE\_RESULTテーブルのGSYS\_EXT\_VCH1フィールドに書き込まれる仮想キュー ID情報の信頼性を示します(表12を参照)。

表 12:信頼性フラグの値

値	値が示す内容
ボイス コール/インタラクション	
1 (ok)	GSYS_EXT_VCH1 フィールドに格納された仮想キュー ID (VQID)が有効。
2 (過去に有効)	GSYS_EXT_VCH1 フィールドに格納された仮想キュー ID (VQID)が過去に有効だった。
0 (不明)	ルーティング関連の通知に仮想キュー ID情報が存在しない。

## 構成に関する推奨事項

この項では、仮想キュー機能に関連したGenesys Configuration Layerの構成設定について説明します。

Interaction ConcentratorとURS両方のリリースが仮想キュー機能をサポートする場合は、デフォルトの構成設定で仮想キューデータを格納できます。

ICON アプリケーション オブジェクト、仮想キュー DN オブジェクト、および交換機オブジェクトの構成設定を使用して、以下のように仮想キューのモニタを操作できます。

- Interaction Concentrator の格納モードを変更する。
- 特定の仮想キューのモニタとデータ格納を無効にする。
- 交換機レベルで(つまり、特定の交換機に属するすべての仮想キューに対して)、モニタとデータ格納を無効にする。

## Universal Routing Server

Interaction Concentratorで仮想キューのモニタを有効にするためには、仮想キュー機能をサポートするURSのリリースが必要ですが、URS側で特別な構成を行う必要はありません。

URS のリリース 7.6 からは、ルーティング要求の AttributeReason を使用して、インタラクションのルーティングに関する追加情報を ICON に提供できます。また、待機しているターゲットに関する情報を URS がインタラクションに添付することもできます (64 ページの「ルーティングポイントのルート結

果のモニタ」を参照)。下流でレポートを行うために、Interaction Concentratorがこの情報を利用するには、URS アプリケーション オブジェクトで、以下の構成オプションを true に設定します。

- report\_reasons
- report\_targets

これらのURS 構成オプションの詳細については、『*Universal Routing 7.6 Reference Manual*』を参照してください。

## ICON アプリケーション

デフォルト設定のまま、ICONが仮想キュー データを受け取り、IDBに格納することができます。

**接続** Interaction Concentratorの仮想キュー モニタを有効にするには、仮想キュー機能をサポートするURSのリリースが必要ですが、ICONはデータをT-Serverから受け取ります。したがって、URSとの接続は必要ありません。

**vq-write-mode** vq-write-mode構成オプションで、必要に応じて仮想キュー データの格納モードを切り替えることができます。格納モードの詳細については、62ページの「格納モード」を参照してください。vq-write-modeオプションは、ICON アプリケーション オブジェクトの[オプション]タブにある callconcentratorセクションで設定します。

**extended-route-result** extended-route-result構成オプションで、ICONがIDBに(URSからの)拡張ルーティング結果を格納するかどうかを指定します。extended-route-resultは、ICONアプリケーション オブジェクトの[オプション]タブにある callconcentratorセクションで設定します。

---

**注：** IDB に拡張ルーティング結果を格納するには、Universal Routing Server (URS) リリース 7.6 と Interaction Server リリース 7.6.000.18 (以上) が必要です。

---

これらのオプションの詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』の構成オプションについての章を参照してください。

## 仮想キュー DN

特定の仮想キューのモニタとデータ格納を無効にする必要がある場合以外は、Configuration Layerで、仮想キューを表すDNオブジェクトの構成を行う必要はありません。

**monitor** monitor構成オプションで、必要に応じて、ICONによる特定の仮想キューのモニタとデータ格納をオフにできます。このオプションを0に設定すると、この仮想キューに関連するイベントを受信するものとしてICONがT-Serverに登録されません。monitorオプションは、特定の仮想キューに対して構成されたDNオブジェクトの[付加情報]タブにあるgtsセクションで設定します。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』でオプションの説明を参照してください。

## 交換機

特定の交換機の仮想キューのモニタとデータ格納を無効にする必要がある場合以外は、Configuration Layerで、対応する交換機オブジェクトの構成を行う必要はありません。

**support-dn-type-5** support-dn-type-5構成オプションで、必要に応じて、特定の交換機に属するすべての仮想キューについて、Interaction Concentratorによるモニタとデータ格納をオフにできます。このオプションを0に設定すると、この交換機に属する仮想キューDNに関連するイベントを受信するものとしてICONがT-Serverに登録されません。この場合、交換機に属するいずれかの仮想キューのmonitorオプションが1に設定されていても、ICONは仮想キュー関連のTEventsの処理や格納を行いません。support-dn-type-5オプションは、特定の交換機に対して構成された交換機オブジェクトの[付加情報]タブにあるgtsセクションで設定します。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』でオプションの説明を参照してください。



# 6

## エージェント状態とログインセッション

Interaction Concentratorのエージェント状態データは、エージェント ログインセッションのコンテキスト内でエージェントのアクティビティに関する詳細な情報を提供します。この章では、エージェント状態とログインセッションに関するデータをIDBで使用可能にする方法を説明します。

この章には以下の項があります。

- [エージェントモデルとログインセッションモデル \(69 ページ\)](#)
- [入手可能なエージェント状態およびログインセッションのデータ \(75 ページ\)](#)
- [後処理と準備中のエージェント状態 \(77 ページ\)](#)

---

## エージェントモデルとログインセッションモデル

この項では、エージェントアクティビティについてのInteraction Concentrator (ICON)データに関する用語、要素(オブジェクト)、モデルについて説明します。

以下の項目について取り上げます。

- エージェントオブジェクトとログインセッションオブジェクト
- エージェント状態モデル ([70 ページ](#)を参照)。
- ログインセッションモデル ([74 ページ](#)を参照)。

## エージェントオブジェクトとログインセッションオブジェクト

以下の用語は、ICONによるレポートデータ格納の対象となるエージェントおよびログインセッションのオブジェクトと要素を表します。

- *ACDQ*—ACD (Automatic Call Distribution : 自動呼分配) キューオブジェクト (ACD デバイスまたは ACD グループ)。

- **エージェント状態** — エージェントに関連付けられたインタラクションに関連したエージェントの状態。音声インタラクションの場合は、さらにACDQに関連したエージェントの状態。

ICON はエージェント状態のイベント レポートによって、エージェント状態の変化に関する情報を入手します。ICON はさまざまなメディア タイプについて個別にエージェント状態を追跡します。

- **ログインセッション**—ACDQまたは特定の交換機の他のエンドポイントにエージェントがログインしている時間。
- **メディアタイプ**—エージェントのインタラクション(電子メール、チャットなど)に適用可能な通信媒体。
- **保留エージェント状態**— **ビジー状態**後にエージェントが移行する**エージェント状態**。
  - ◆ エージェントが任意の状態からビジーに移行すると、ICON が移行前の状態を保留状態として保持する。
  - ◆ エージェント状態がビジーのときに、ICON がエージェント状態の変化に関する情報を受け取ると、ICON が新しい状態を保留状態として保持する。
  - ◆ ビジー状態が終了すると、ICON がエージェント状態を保留状態から復元する。
- **理由**— エージェント状態が変化した理由。

ボイス コールの場合は、以下のソースの T-Server レポートから、ICON がエージェント状態の理由を入手します。

- ◆ workmode パラメータ
- ◆ TEvent属性拡張(ハードウェア理由)、事前定義されたReasonCodeキーの値から取得
- ◆ TEvent 属性理由 (ソフトウェア理由)、キーと値のペアの値から取得

マルチメディアの場合、Interaction Server のマルチメディア レポート イベントに、エージェント状態の変化の理由に関する情報を持つ属性が含まれる場合があります。ICON は、ハードウェア理由を格納する場合と同じ方法で、この理由コード情報を格納します。Interaction Server がレポートするエージェント状態に関連するイベントには、1 つの理由だけが含まれません。

## エージェント状態モデル — 音声およびマルチメディア

### エージェント優先の エージェント状態 モデル

Interaction Concentrator はエージェント優先のエージェント状態モデルをサポートします。このモデルでは、エージェントが関連付けられたエンドポイントやACDQオブジェクトの数には関係なく、エージェント ログインセッション内の各メディア タイプ(音声、電子メール、またはチャット)に対して、交換機機能がエージェントの1つの状態を維持します。各メディア タイプに関して、エージェントが異なる状態を持つことができます。

### エージェント状態

表13に、ボイス コールに関連するエージェント状態を示します。ACDQはマルチメディアには適用されませんが、マルチメディア メディア タイプのエー

エージェント状態がこれに相当します。表13には、各状態への移行を表すT-Server またはInteraction Server (MM)レポート イベントも示されています。

表 13: エージェント状態および状態移行イベント

エージェント状態	説明/状態移行イベント
Null	<p>エージェントが特定のデバイスのACDQにログオンしていない。ACDQにログオンすると、この状態から移行する。同様に、ACDQからログオフすると、この状態に移行する。</p> <p><b>TEvent</b></p> <ul style="list-style-type: none"> <li>• EventAgentLogout— エージェントがログオンしたすべての ACDQ オブジェクトについて、エージェント状態の変化をレポートする。</li> <li>• EventQueueLogout— 特定のACDQについて、エージェント状態の変化をレポートする。</li> </ul> <p><b>MMイベント</b></p> <ul style="list-style-type: none"> <li>• EventAgentLogout— エージェントがログオンした作業場所のすべてのエンドポイントとメディアタイプについて、エージェント状態の変化をレポートする。</li> <li>• EventMediaRemoved— 指定のメディアタイプについて、エージェントがもうログオンしていないことをレポートする。</li> </ul>
Login	<p>エージェントが特定のデバイスのACDQにログオンし、ACDキューのアクティビティに関する準備ができています。この状態はエージェントがACD呼を受け入れる準備ができたことを意味しない(Readyを参照)。</p> <p><b>TEvent</b>      EventAgentLogin  <b>注:</b> TEventにThisQueueパラメータがあるかどうかで、エージェントがACDキューにログオンしているかどうかを指定する。</p> <p><b>MMイベント</b>      EventAgentLogin  <b>注:</b> このイベントにメディアタイプについての情報が含まれる場合は、メディアタイプごとに個別のレコードが作成され、メディアタイプごとに適切なエージェント状態が設定される。</p>
NotReady	<p>エージェントが特定のデバイスのACDQにログオンしているが、ACDキューから配信されるインタラクションを処理する準備はできていない。この状態のエージェントは、ACD呼以外の呼を受信可能。</p> <p><b>TEvent</b>      EventAgentNotReady。AgentAfterCallWorkと等しくないworkmodeパラメータを持つ。</p> <p><b>MMイベント</b></p> <ul style="list-style-type: none"> <li>• EventNotReadyForMedia</li> <li>• EventMediaAdded— エージェントのログインセッションに追加されたメディアタイプについての情報を提供。</li> </ul> <p><b>注:</b> ICONはエージェント状態についての情報をイベント属性から取得する。マルチメディアの場合は、EventMediaAddedイベントに対して指定されるエージェント状態は常にNotReady。</p>

表 13: エージェント状態および状態移行イベント (つづき)

エージェント状態	説明/状態移行イベント
Ready	<p>エージェントが特定のデバイスのACDQにログオンし、ACD呼以外を処理している場合でも、ACDインタラク션을処理する準備ができています。</p> <p>TEvent            EventAgentReady</p> <p>MMイベント      EventReadyForMedia</p>
Busy	<p>エージェントが特定のデバイスのACDQにログオンし、デバイスの既存の呼に関与している。デバイスの呼が保留中の場合も含まれる。</p> <p>このエージェント状態には特に移行イベントは存在しない。この状態は、エージェントがインタラク션に関与してから、エージェントが切断するまでの期間をカバーする。</p> <p>ACDインタラク션に加えて、エージェント間の通話、スーパーバイザとエージェント間の通話、およびプライベート通話でも、ビジー状態への移行が発生する。</p>
ACW	<p>エージェントがもう呼に接続していないが、まだ前の呼に関連する作業に従事している(たとえば、業務用注文フォームの更新などの管理作業)。この状態では、エージェントがACD呼を受信できないが、場合によってはACD以外の呼は受信可能。</p> <p><b>注:</b> ICONはT-Serverイベントから呼情報を取得することはできない。ICONは、ビジー状態から他の状態へ移行する前のエージェントのデバイスで表される最後の呼に関する情報を保持する。ビジー状態からACW状態に移行する場合は、ICONがエージェントのデバイスの最後の呼をACW状態に関連するものとして割り当てる。</p> <p>TEvent            EventAgentNotReady、AgentAfterCallWorkと等しいworkmodeパラメータを持つ。</p> <p>MMイベント      適用されない。</p>
Unknown	<p>エージェントのログインセッションが存在するが、ICONがエージェント状態についての情報を持っていない(たとえば、T-Serverからの切断が原因)。</p>

**注:** Interaction Concentrator リリース 7.5 以降では、カスタムのエージェント状態もサポートされます。詳細については、91 ページの第 8 章を参照してください。

#### エージェント状態の FSM

図3に、ログインセッション内のボイスコールに対するエージェント状態のFSM (Finite State Machine: 有限状態機械)を示します。AfterCallWork状態と状態移行イベントの名前を除き、マルチメディアのエージェント状態のESMも同様です。

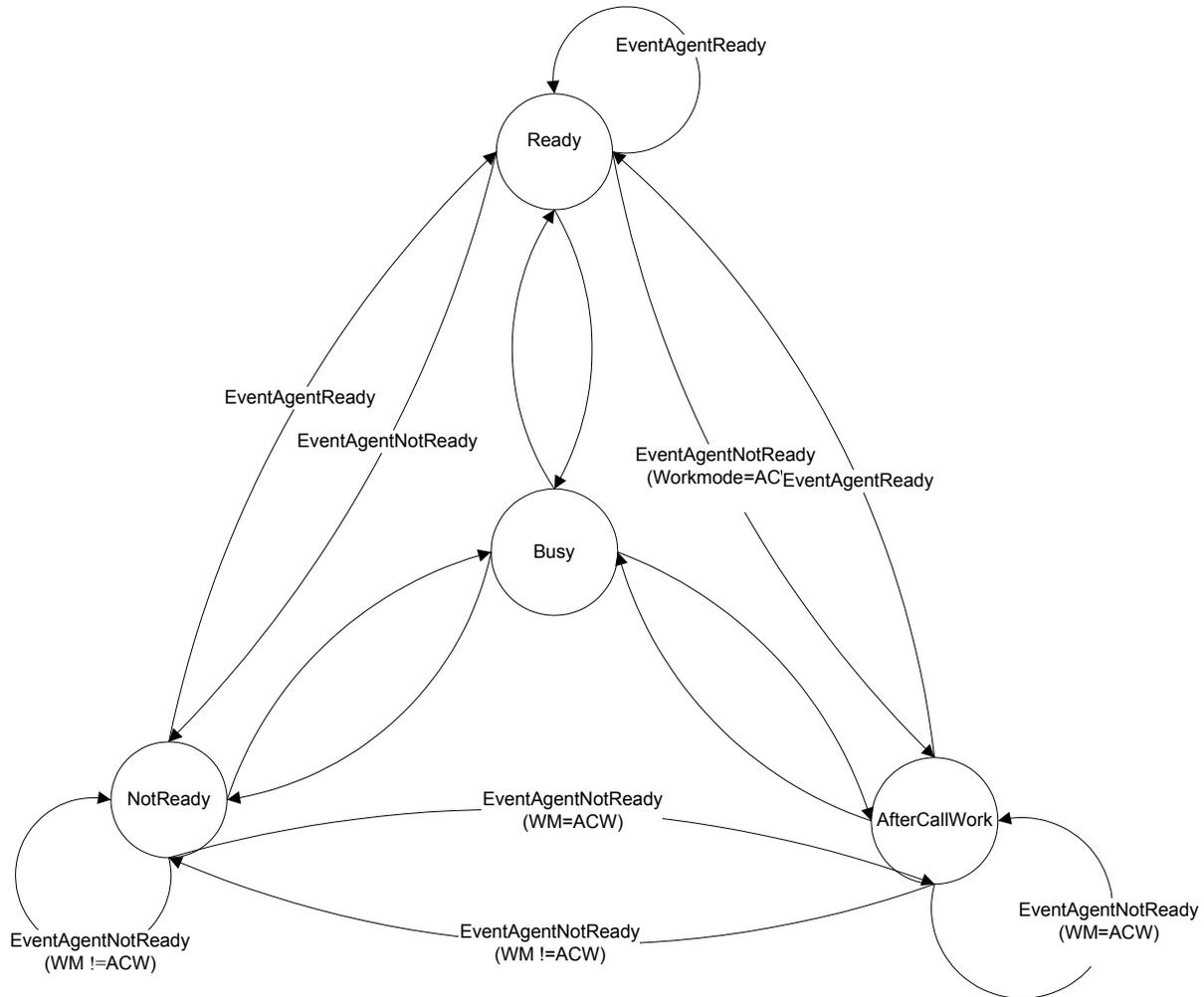


図3: エージェント状態のFSM

### エージェント状態の 復元

起動時または再接続時に、ICONがT-Serverに登録され、エージェント状態のモニタを開始する際に、EventRegistered TEventイベントがICONにボイスコールの場合の現在のエージェント状態のスナップショットを提供し、さらに、T-Serverが配信しエージェントのデスクトップに表示された呼のリストも提供します。

マルチメディアソリューションの場合は、ICONがInteraction Serverに登録され、エージェント状態のモニタを開始する際に、EventPlaceAgentStateレポートイベントが同様にICONに、さまざまなメディアに対する現在のエージェント状態のスナップショットおよびInteraction Serverが配信しエージェントのデスクトップに表示されたインタラクションのリストを提供します。

## エージェント状態モデル—SIP チャット

Interaction Concentratorは、SIP Serverから受信するエージェント状態関連イベントのチャット メディア タイプを処理します。これらのイベントから抽出されたデータは、既存のIDBテーブルに格納されます。メディア タイプは、これらのテーブル内のGSYS\_EXT\_INT1フィールドに整数値コードとして格納されます。

**状態関連イベント** Interaction Concentratorは各メディア タイプ内で簡単なエージェント モデルをサポートします。各メディア タイプについて、DNのエージェントが異なる状態を持つことができます(たとえば、音声についてはエージェント状態がビジーだが、チャットについては待機中)。

SIP Serverはエージェント状態関連イベント(EventAgentLogin、EventAgentReady、EventAgentNotReady、EventAgentLogout)でメディア タイプについての情報は提供しません。したがって、エージェント状態関連イベントからメディア タイプを判断することができないため、ICONは、エージェントのログインセッションに関連付けられたすべてのメディア タイプに適用可能なものとして、受け取ったイベントの情報を処理します。

**パーティ状態の変化** Interaction Concentratorは、通話パーティに関連する呼からメディア タイプを判断できません。ICONは、このメディア タイプに作成されたエージェント状態に対してのみ、パーティ状態の変化を処理します。他のメディア タイプのエージェント状態は、このパーティ状態変化によって影響を受けません。

**DND 状態変化** SIP Serverは、EventDNDOnおよびEventDNDOffイベントでメディア タイプについての情報を提供しません。Interaction Concentratorは、メディア タイプに関係なく、DND状態の変化に関する情報を処理し格納します。DNに複数のメディア タイプが構成されていて、メディア タイプごとにエージェント状態が異なる場合は、関連するレコードのエージェント状態はLoggedInになります。

**ThisQueue** Interaction Concentratorは、音声メディア インタラクションについて、ThisQueue属性についての情報を処理し格納します。DNの音声メディア タイプが無効な場合は、チャットメディア タイプに対してこの属性が処理されます。

## ログインセッションモデル

エージェントのログインセッションが開始するのは、エージェントが交換機にログオンしていなかった期間、またはエージェントに関する情報を入手できなかった期間(構成可能)があり、その後で、ICONが初めてエージェントについてEventAgentLogin イベントを受信したときです(『Interaction Concentrator 7.6 Deployment Guide』のgls-max-inactivity構成オプションの説明を参照)。

エージェントがすでに交換機にログオンしているときに、追加のACDQまたはエンドポイントオブジェクトにログオンするか、メディア タイプが追加または削除された場合、ICONは既存のログインセッションを続行します。

エージェントのログインセッションが終了するのは、エージェントがもう ACDQ にログオンしていないとき、または設定されたアクティビティなしの最大時間内に、エージェント関連のアクティビティがなかったときです。

図4は、ログインセッションのFSMを示しています。

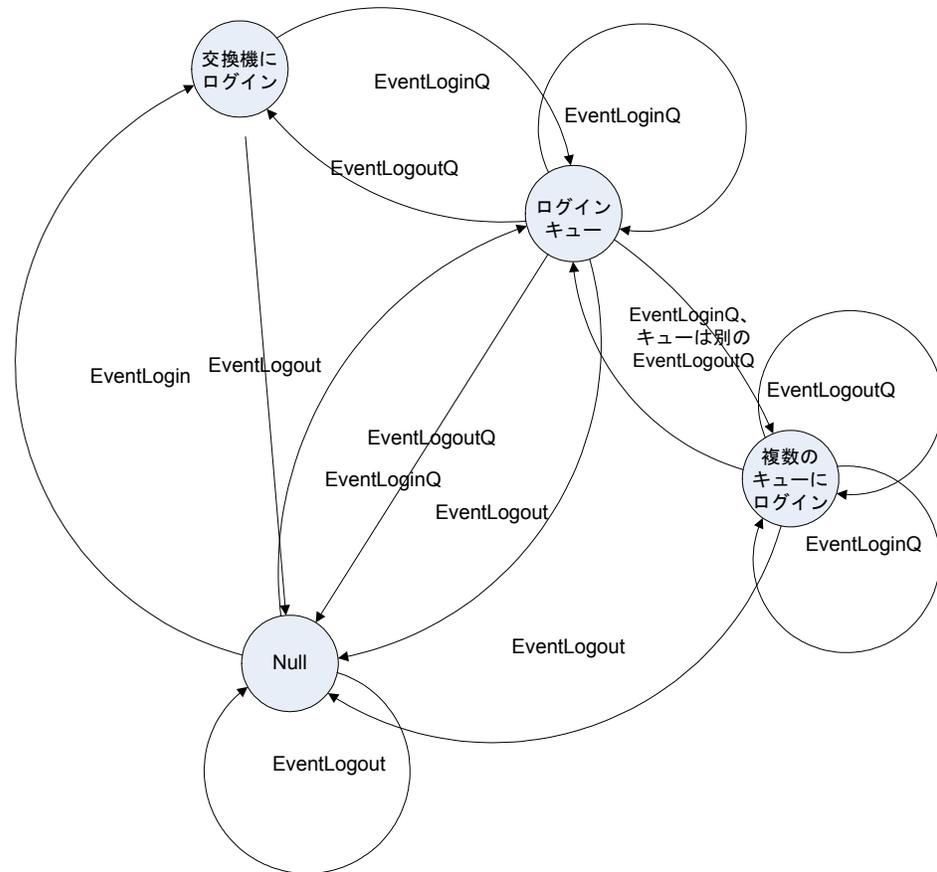


図4: ログインセッションの有限状態機械

## 入手可能なエージェント状態およびログインセッションのデータ

この項では、ICON が収集するエージェント状態データとログインセッションデータについて説明します。ただし、この役割を実行するように ICON が構成されている必要があります。

ICON がエージェント状態とログインセッションについてのデータを格納する IDB テーブルの概要については、31 ページの「エージェント状態関連およびログインセッション関連テーブル」を参照してください。詳細については、ご使用の RDBMS について、『*Interaction Concentrator 7.6 Physical Data Model*』を参照してください。

## エージェント状態およびログインセッションのデータ

ICONは、ログインセッションとエージェント状態の変化について、実際のデータと履歴データの両方を格納し、さらに、エージェント状態、ログインセッション、およびエンドポイントの関連付けも格納します。ICONはさまざまなメディアタイプについて個別に変化を追跡します。

以下のデータを入手できます。

- エージェントログインセッションについての現在の情報と履歴情報
- ログインセッションとエンドポイント間の関連付けについての現在の情報と履歴情報

---

**注：** マルチメディア レポート イベントには、エージェントの作業場所についての情報が含まれますが、エージェント エンドポイント DN についての情報は含まれません。したがって、エージェント アクティビティ関連テーブルの ICON レコードには、マルチメディアの場合、有意義なエンドポイント情報は含まれません。

---

- エージェント ログイン セッション中のエージェント状態の変化についての詳細情報
  - ◆ エージェント状態、保留状態、workmode、ハードウェア理由コードの変化
  - ◆ 状態変化の時刻
  - ◆ 他のパーティの接続と切断
- エージェント ログイン セッション中のハードウェアおよびソフトウェア理由コードの変化についての詳細情報

---

**注：** マルチメディア レポート イベントにエージェント状態の変化の理由コードに関する情報が含まれる場合は、音声の場合に T-Server レポートが提供するハードウェア理由コードを格納するのと同様の方法で、ICON が理由コードを格納します。ハードウェア理由コードはマルチメディアには適用されません。

---

- エージェント ログイン セッション内での DND 機能利用に関する詳細情報。音声の場合、個々の DN に対して DND をアクティブ化/非アクティブ化可能。マルチメディアの場合、作業場所全体に対してのみ DND をアクティブ化/非アクティブ化可能

## 事前計算されたエージェント状態メトリクス

生のオブジェクトデータに加えて、ICON では以下のエージェント状態関連の統計も格納できます。

- 以下に示すエージェント状態の持続時間
  - ◆ Duration\_ready
  - ◆ Duration\_notready

- ◆ DurationACW (マルチメディアには適用されない)
- ◆ DurationBusy
- 以下に示すエージェントのworkmodeの持続時間(マルチメディアには適用されない)
  - ◆ Duration\_UNKNOWN
  - ◆ Duration\_AUX
  - ◆ Duration\_LegalGuard
  - ◆ Duration\_GoAway
  - ◆ Duration\_ReturnBack

---

## 後処理と準備中のエージェント状態

ACW (After-Call Work : 後処理)は、インバウンド呼の直後にエージェントが行う必要がある作業で、データ入力やトランザクションを完了するための内線電話などが含まれます。このモード中は、エージェントが別のインバウンド呼を受けることはできないとみなされます。この項では、Interaction Concentratorの後処理と準備中のエージェント状態の機能について説明します。

### 中断のない ACW または準備中の持続時間

Interaction Concentratorには、後処理やNotReadyエージェント状態を継続する機能があり、後処理や準備中のときにエージェントが電話をかけたり受けたりすることによりこれらが中断することはありません。以下のいずれかが発生すると、後処理またはNotReadyエージェント状態が完了したとICONが認識します。

- エージェントがログアウトする。
- エージェントが自身を Ready モードにする。
- 後処理以外の理由でエージェントがNotReadyになる(たとえば、エージェントが一定時間デスクから離れるなどの間接的な workmode の変化)。

中断なしのACW またはNotReadyエージェント状態の機能を有効にするには、交換機の [付加情報] タブで、gls-enable-acw-busy構成オプションをfalseに設定します。

### 中断のある ACW または準備中の持続時間

Interaction Concentratorには、エージェントがACWまたはNotReady期間中に電話をかけたり受けたりしたときに、後処理や準備中のエージェント状態を中断する機能もあります。デフォルトでは、エージェントが別の呼を処理しているときには、ACWやNotReadyエージェント状態が中断します。中断を有効にするには、交換機の [付加情報] タブでenable-acw-busy構成オプションをtrueに設定します。このオプションの設定の詳細については、『Interaction Concentrator 7.6 Deployment Guide』の構成オプションの章を参照してください。

## ACW とインタラクションの関連付け

Interaction Concentratorでは、後処理開始の直前の音声インタラクション(最初の音声インタラクション)と後処理を関連付けたり、最後のインタラクションと後処理を関連付けることができます(後者がデフォルト)。

前者の場合、後処理中に発生した後続の音声インタラクションは、ACW処理と関連しているとみなされ、ACW関連のメトリクス測定は中断されません。

この機能を有効にするには、交換機レベルまたはICONアプリケーションで `gls-acw-first` 構成オプションを設定します。ICONアプリケーションで設定した値は、SIP交換機を除くすべての交換機に適用されます。SIP交換機の場合は、交換機レベルでオプションを指定する必要があります。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』の構成オプションの章を参照してください。



# 7

## Outbound Contact との統合

この章では、Genesys Outbound Contact ソリューションのデータを Interaction Database で使用可能にする方法を説明します。

この章には以下の項があります。

- [Outbound オブジェクトとモデル \(79 ページ\)](#)
- [入手可能な Outbound データ \(82 ページ\)](#)
- [Outbound Contact の構築シナリオ \(84 ページ\)](#)
- [構成に関する推奨事項 \(87 ページ\)](#)

---

### Outbound オブジェクトとモデル

この項では、Genesys Outbound Contactに関連する用語、要素(オブジェクト)、およびモデルについて説明します。

#### Outbound Contact オブジェクト

以下にOutbound Contactのオブジェクトと要素についての用語を示します。これらについて、Interaction Concentrator (ICON)がレポート データを格納します。

- **キャンペーン**— 顧客データの管理、呼の生成、呼の結果のモニタと処理のためのマスタ プラン。Outbound Contact がキャンペーンを使用してアウトバウンド発信を自動化します。
- **キャンペーングループ**— キャンペーン、コーリング リスト、およびキャンペーン アクティビティを処理するために割り当てたリソースのグループ (エージェントなど) の間のランタイム関連付け。
- **コーリング リスト**— 顧客データ (場合によっては、発信フィルタを適用) を含むレコードからなるデータベース テーブル。
- **レコード**— 顧客データの基本単位。各レコードが 1 つの顧客電話番号を表します。

- フィールド—レコード内の1つのデータ(電話番号など)。コーリング リストには、Outbound Contact ソリューションが機能するために必要な必須のフィールドが含まれています。コーリング リストにカスタム フィールドを含めることもできます。
- チェーン—論理的に結合されたレコード(通常は同じ顧客のコンタクト情報を表す)。たとえば、同じ顧客の自宅、職場、携帯の電話番号の3つのレコードがチェーンを構成する場合などです。
- フォーマット—コーリング リストのデータをまとめたテンプレート。コーリング リストが作成されレコードが生成されると、レコードの顧客データがフィールドに入力され、システム デフォルト設定またはユーザ定義の設定に従って、フォーマットで設定されたフィールドプロパティが適用されます。

**注：** Outbound Contact オブジェクトの詳細については、『*Outbound Contact 7.6 Deployment Guide*』を参照してください。

## キャンペーン モデル

図5に、キャンペーン グループのFSM (Finite State Machine : 有限状態機械) を示します。

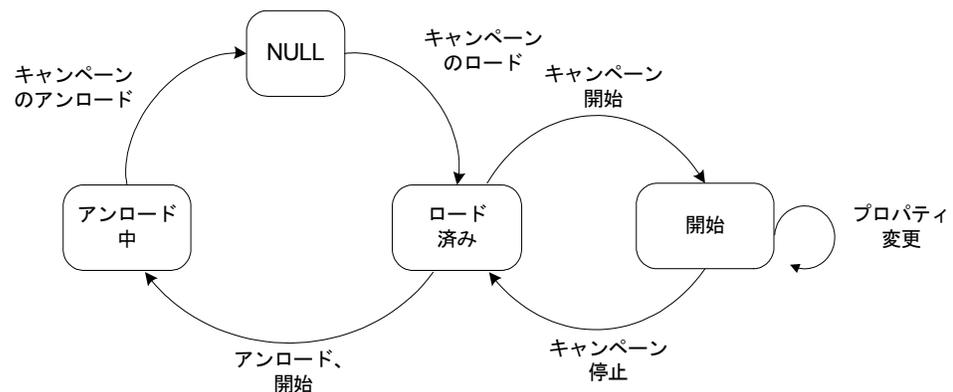


図5: キャンペーン グループの有限状態機械

## チェーン モデル

図6に、チェーンの場合のFSMを示します。

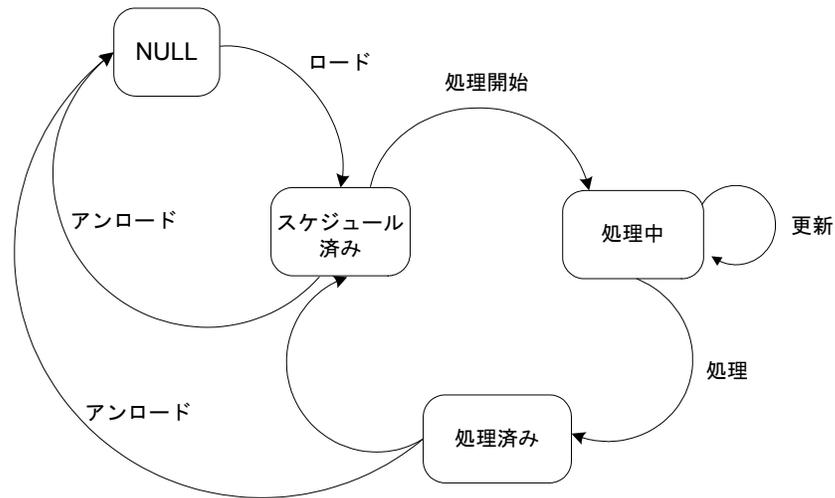


図6: チェーンの有限状態機械

**チェーン状態** 図6には、以下のようなチェーン状態が示されています。

- *NULL*— 初期状態。データベースからこのチェーンのレコードがまだ選択されていません。これは最終状態でもあり、チェーンのすべてのレコードがファイナライズまたはアンロードされ、データベースで更新されます。
- *処理中*— チェーンのレコードの処理が開始し、処理のためのリソースが割り当てられている状態。
- *処理済み*— レコードが処理された状態。実際の処理結果に応じて、チェーンのレコードがファイナライズまたは再スケジュールされます。
- *スケジュール済み*— コーリングリストからチェーンがロードされたため、または前の処理の試みにより、チェーンのレコードが処理のためにスケジュールされた状態。指定した日付と時刻に処理する、または今すぐ (Outbound Contact Server (OCS) がチェーン処理のリソースを入手したらすぐに) 処理するようにチェーンをスケジュールできます。

---

## 入手可能な Outbound データ

この項では、OCSとICONの両方が正しく構成されていることを前提に、ICONがOCSと通信する方法、OCSがICONに送信できるデータの種類について説明します。

### ICON と OCS の通信

OCSが通常のクライアント接続のために開くポートを使用して、ICONがOCSに接続します。他のOCSクライアントとは異なり、ICONは特別なプロトコルを使用してレポート関連のデータを受信します。OCSデータには、OCSオブジェクトに関するランタイム情報と、OCSが特にレポートのために計算する特定の統計が含まれます。

所定のコンタクトセンターで複数のOCSインスタンスが同時に実行されている場合、1つのICONインスタンスを使用して、1つのOCSインスタンスまたは複数のOCSインスタンスからのデータを処理し格納することができます。

### Outbound オブジェクト データ

ICONにはキャンペーン情報とチェーン情報が格納されます。データが入手可能になると、所定のキャンペーンセッション内の履歴および呼とパーティ間の関連付けの両方が格納されます。

特に、アウトバウンドレポートのためにIDBに以下のデータを格納するために、ICONとOCSを統合できます。

- キャンペーン構成プロパティの変更履歴および最適化パラメータのターゲット値。
- チェーン処理の履歴と結果。
- チェーン処理中にOCSがコーリングリストレコードの加えた変更履歴およびICONの追跡対象の変更履歴。
- アウトバウンド呼生成の試みの結果 (成功または失敗)。
- CPD (Call Progress Detection) の呼の結果。
- エージェントが割り当てた呼の結果。CPDの呼の結果とは個別に格納され、CPDの結果を上書きしない。

---

**注：** アウトバウンドデータを格納する IDB テーブルすべてのリストと説明については、ご使用の RDBMS について、『*Interaction Concentrator 7.6 Physical Data Model*』を参照してください。

---

## 事前計算された OCS メトリクス

OCSを構成して、生のオブジェクトデータに加えて、事前計算されたメトリクスとも呼ばれるアウトバウンド関連の統計をOCSで計算し、ICONで格納することができます。

表14に、OCSがICONに提供し、ICONがGO\_METRICS IDBテーブルに格納する事前計算されたメトリクスを示します。

表 14:事前計算された OCS メトリクス

メトリクス	説明
現在アクティブなすべてのキャンペーングループおよびアクティブなキャンペーングループに関与するすべてのコーリングリストについて、OCSが以下のメトリクスのサブセットを計算する。OCSは10分ごとに1つのデータパケットとして、これらのメトリクスをICONに配信する。	
コーリングリスト当たりのレコードの合計数	コーリングリストデータベースビューの物理レコードの数(たとえば、データベーステーブルのレコード、および場合によっては発信フィルタのWHERE句)。
キャンペーングループ当たりのレコードの合計数	キャンペーングループに関与するキャンペーンにコーリングリストが割り当てられているすべてのデータベースビューの物理レコードの数。
コーリングリスト当たりのチェーンの合計数	コーリングリストデータベースビューの論理チェーンの数(たとえば、データベーステーブルのレコード、および場合によっては発信フィルタのWHERE句)。論理チェーンとは、同じチェーンID値を持つデータベーステーブルレコード。
キャンペーングループ当たりのチェーンの合計数	キャンペーングループに関与するキャンペーンにコーリングリストが割り当てられているすべてのデータベースビューの論理チェーンの数。
コーリングリスト当たりの現在未処理のレコード数	コーリングリストデータベースビューの物理レコードのうち、タイプGENERALと状況READYを持つものの数(たとえば、データベーステーブルのレコード、および場合によっては発信フィルタのWHERE句)。
キャンペーングループ当たりの現在未処理のレコード数	すべてのデータベースビューのレコードのうち、タイプGENERALと状況READYを持つものの数(キャンペーングループに関与するキャンペーンにコーリングリストが割り当てられている場合)。
コーリングリスト当たりの現在未処理のチェーン数	コーリングリストデータベースビューの論理チェーンのうち、チェーン内の最低1つのレコードがタイプGENERALと状況READYを持つものの数(たとえば、データベーステーブルのレコード、および場合によっては発信フィルタのWHERE句)。
キャンペーングループ当たりの現在未処理のチェーン数	すべてのデータベースビューの論理チェーンのうち、チェーン内の最低1つのレコードがタイプGENERALと状況READYを持つものの数(キャンペーングループに関与するキャンペーンにコーリングリストが割り当てられている場合)。

表 14:事前計算された OCS メトリクス (つづき)

メトリクス	説明
キャンペーングループ当たりの現在ビジーのダイヤルポートの数	CPD Serverポートまたは交換機の呼分類ポートのうち、現在ビジー状態のポートの数(発信中のアウトバウンド呼で占有されているポート)。
キャンペーングループ当たりの現在使用中(ビジー)のポートの数	CPD Serverがすでに発信した呼によって現在使用中のポートの数。 <b>注:</b> OCSがこのメトリクスを計算するのは、CPD Serverについて、ASM (Active Switching Matrix) 発信モードでアクティブにされたキャンペーングループの場合のみ。
オーバーダイヤルされたアウトバウンド呼	Outbound Dialing Engineが所定の呼をオーバーダイヤルだとみなすことを示す。 オーバーダイヤルの呼については、Outbound Contactのマニュアルを参照。 <b>注:</b> 現在アクティブなキャンペーングループすべてについて、OCSがこのメトリクスを計算する。呼がオーバーダイヤルになると、OCSがこのメトリクスをICONに配信する。OCSはこのメトリクスを事前に設定された間隔では配信せずに、オーバーダイヤルの呼が発生するたびに計算してICONに渡す。
現在アクティブなすべてのキャンペーングループについて、OCSが以下のメトリクスのサブセットを計算する。発信が成功するたびに、OCSが1つのスナップショットイベントの形で、ICONにこれらのメトリクスを配信する。OCSはこれらのメトリクスを事前に設定された間隔では配信せずに、アウトバウンド呼の発信が成功するたびに計算してICONに渡す。	
アウトバウンド呼のダイヤル時間	ダイヤルの継続時間(ミリ秒単位)。
アウトバウンド呼の転送時間	呼の転送の継続時間(ミリ秒単位)。
CPD時間	接続後のCPDの継続時間(ミリ秒単位)。

## Outbound Contact の構築シナリオ

Interaction ConcentratorはあらゆるタイプのOutbound Contactの構築をサポートします。構築シナリオに関わらず、一貫してアウトバウンドデータを格納することができます。

この項では、以下の2つの主要な構築シナリオを取り上げます。

- シングルサイト構築。1つのICONインスタンスが専用に出バウンドデータを処理します。
- マルチサイト構築。すべてのサイトに対して1つのOCSインスタンスがあり、各サイトに1つのICONインスタンスがあります。

**図の表記規則** 構築の図を簡潔にするために、この項では以下の表記規則を採用します。

- 実際の構築では必要だが、図では ICON と IDB 間の接続を可能にする DB Server は省略 (20 ページの図 1 を参照)。
- 実際の構築では必要だが、構成データの格納は図示しない。

## シングル サイト構築の例

シングル サイト コンタクト センターに最適な最も簡単な構築シナリオは、1 つの ICON インスタンスから構成され、これが専用としてすべてのアウトバウンドデータを 1 つの IDB インスタンスに格納します(図 7 を参照)。

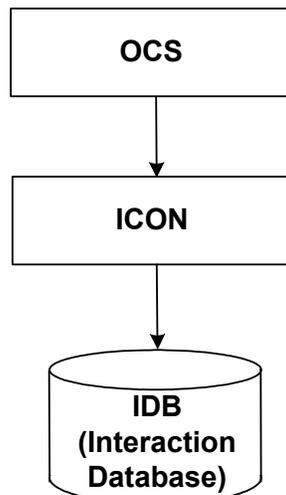


図 7: シングル サイト構築の例: 1 つの OCS インスタンスと専用 ICON インスタンス

このシナリオでは、ICON が専用として、オブジェクト情報と事前計算されたメトリクス両方のアウトバウンドデータを格納します。この ICON インスタンスは T-Server との接続または Interaction Server との接続 (マルチメディア ソリューションの場合) を持ちません。また、T-Server や Interaction Server からインタラクションデータを収集しません。別の ICON インスタンスが、CTI 関連データおよびインタラクションデータを同じ IDB または別の IDB に格納するものとみなされます。

## マルチサイト構築の例

最も簡単なマルチサイト構築シナリオは、さまざまなコンタクトセンター サイトで実行される ICON インスタンスから構成されます。これらすべての ICON インスタンスが、同じ OCS インスタンスから受信するアウトバウンドデータを格納し、さらに、それぞれの ICON インスタンスが特定の T-Server または Interaction Server から受信する CTI データを格納します。

図 8 に、簡単なマルチサイト構築シナリオを示します。図を簡潔にするために、ボイス コール用の構築だけを示し、IDB の構築は省略します。マルチメディア ソリューションの場合の OCS インスタンスと Interaction Server との関係は、音声ソリューションの場合の OCS インスタンスと T-Server との関係と同様です。

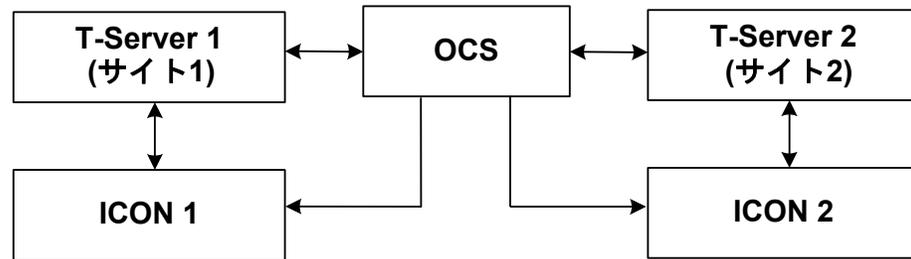


図 8: マルチサイト構築の例 : 1 つの OCS インスタンスと複数の ICON インスタンス

このシナリオでは、以下のように処理が実行されます。

- ICON 1 が格納するデータ :
  - ◆ OCS がサイト 1 だけで生成するアウトバウンド アクティビティ関連データ
  - ◆ サイト1用のT-Server 1がレポートする呼アクティビティ関連のCTIデータ
- ICON 2 が格納するデータ :
  - ◆ OCS がサイト 2 だけで生成するアウトバウンド アクティビティ関連データ
  - ◆ サイト2用のT-Server 2がレポートする呼アクティビティ関連のCTIデータ

サイト1で発生した呼がサイト2のアウトバウンド エージェントによって処理されている場合は、ICON 1がアウトバウンド固有のデータをレポートし、ICON 2がこの呼に関連付けられた音声(CTI)アクティビティをレポートします。

## IDB のアウトバウンド データと CTI データの照合

同じインタラクションのCTIまたはインタラクション データ(T-Serverまたは Interaction Serverがレポート)とアウトバウンドデータ(OCSがレポート)を照合するには、呼の試行 ID(CallAttId) を使用します。ICON がこれを GOX\_Chain\_Callテーブルに格納します。

**注：** この機能は Outbound Contact Server (OCS) リリース 7.2 では使用できません。

## OCS レポートの制限

キャンペーン グループが以下の発信モードで実行されている場合は、OCS が特定のタイプのアウトバウンド呼を追跡できません。

- プレビュー モード—Agent Desktop によって発信された呼
- Power GVP モード—GVP Dialer によって発信された呼
- プッシュ プレビュー モード—Interaction Server から Agent Desktop へプッシュされた呼

このようなタイプの呼のチェーン関連イベントには、チェーン処理中に発信されたアウトバウンド呼の一意のID(呼のGUID)は含まれません。したがって、

ICONがGOX\_Chain\_Callテーブルにレコードを作成するために必要な情報を受け取ることができません。

**解決策** 解決策として、呼の試行GUIDを使用して、アウトバウンドチェーンアクティビティについてのデータとテレフォニーアクティビティについてのデータを橋渡しすることができます。OCSがチェーン関連イベントで呼の試行GUIDを通知し、T-ServerとInteraction Serverがアウトバウンド呼のユーザデータにこのGUIDを挿入します。

---

## 構成に関する推奨事項

レポートのためにアウトバウンドデータをIDBに格納するには、Genesys Configuration Layerの特定のアウトバウンド関連の構成オブジェクトとICONアプリケーションオブジェクトで構成の設定を行う必要があります。

## Outbound Contact

OCSがコーリングリストレコードのフィールドの内容に関するデータを処理し、ICONに送信するためには、特別な構成が必要です。

### フィールドオブジェクト

レコード内の1つのデータ(電話番号など)を記述するフィールド構成オブジェクトのそれぞれについて、該当するデータをIDBに格納するために、icon\_attribute オプションを設定する必要があります。

このオプションを設定するには、以下の手順を行います。

1. Configuration Manager のメイン ウィンドウに移動します。
2. 特定のフィールド構成オブジェクトの [ プロパティ ] ダイアログ ボックスを開きます。
3. [ 付加情報 ] タブをクリックします。
4. default という名前の新しいセクションを作成します(存在しない場合)。
5. このセクション内に、icon\_attribute という名前の新しいオプションを作成します。
6. オプションを以下のいずれかの値に設定します。
  - ◆ 1—OCS の必須フィールドを GO\_RECORD テーブルに、カスタム定義のフィールドを GO\_CUSTOM\_FIELDS テーブルに、フィールドの変更履歴を GO\_FIELDHIST テーブルに格納します。
  - ◆ 2—データを保護されたフィールドとして、特別な IDB テーブルである GO\_SECURE\_FIELDS と GO\_SEC\_FIELDHIST に格納します。

このオプションを設定しない場合、または値を0(ゼロ)に設定した場合は、このフィールドオブジェクトに関連したデータをOCSが処理しないため、ICONがこのデータを受け取りません。

詳細については、Outbound Contact 7.6のマニュアルを参照してください。

## OCS アプリケーション オブジェクト

上述の項で述べたコーリング リストのフィールドから、データを選択して送信するために必要なOCSアプリケーション オブジェクトのそれぞれについて、send\_attribute オプションを設定します。詳細については、『*Outbound Contact 7.6 Deployment Guide*』のOutbound Contactの構成オプションについての章で、フィールド レベル オプションに関する項を参照してください。

## キャンペーン グループ オブジェクト

キャンペーン グループに関連付けられたすべてのアクティビティ (チェーン アクティビティを含む)のレポートを有効にするには、キャンペーン グループ オブジェクトの構成プロパティで、有効な音声転送先DNが指定されていることを確認します。このDNは、OCSが接続されるT-Serverがサービスを提供する交換機に存在し、T-Serverが、交換機と接続されるCTIリンクを持っている必要があります。

## ICON アプリケーション

ICONがOCSデータを受信してIDBに格納するためには、ICONと適切なOCSインスタンスとの接続を設定し、関連する構成オプションを設定する必要があります。

### 接続

- 1つのOCSインスタンスを使用する環境では、ICONアプリケーション オブジェクトの [ 接続 ] タブに、ocs アプリケーション オブジェクトを追加します。
- 複数の OCS インスタンスを使用する環境では、構築トポロジを決定する必要があります。具体的には、1つのICON インスタンスですべてまたはいくつかのOCS インスタンスからのデータを処理するか、各OCSに専用のICON インスタンスを割り当てるかを決定します。この決定に基づいて、OCS インスタンスからのデータを格納する必要がある ICON アプリケーション オブジェクトの [ 接続 ] タブに、いくつかの ocs アプリケーション オブジェクトを追加します。

## アウトバウンド関連オプション

### 役割構成オプション

アウトバウンドデータを格納するICONインスタンスそれぞれについて、ICONアプリケーション オブジェクトの [ オプション ] タブで、role オプションの値のリストにgosが含まれていることを確認します。コンタクトセンター全体に1つのICONインスタンスが構築されている場合は、デフォルト値(all)のまま

まにします。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のroleオプションの説明を参照してください。

異なるタイプのデータを異なるIDBに格納する場合は、適切なDAP (Database Access Point)の[オプション]タブでも、roleオプションにgosが指定されていることを確認します。ICONインスタンスがアウトバウンドデータをIDBに格納するために使用するDAPのアプリケーション オブジェクトにある[オプション]タブで、このオプションを設定します。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のroleオプションの説明を参照してください。

### OCS 固有のオプション

以下のICON 構成オプションで、ICON が格納するアウトバウンドデータと格納方法を指定できます。

- gos-write-duplicate-metrics
- gos-write-metrics
- gos-write-metrics-only

『*Interaction Concentrator 7.6 Deployment Guide*』を参照して、gos-write 構成オプションの説明と値について確認してください。ご使用の環境に適切な値を選択し、ICONアプリケーション オブジェクトの[オプション]タブで関連する構成を変更する必要があります。

## マルチテナント環境

マルチテナント構築の場合は、環境テナントの[付加情報]タブに、以下の構成オプションを追加します。

- icon\_attribute—データを保護されたテーブルに格納するか、保護されていないテーブルに格納するかに応じて、1 または 2 に設定します (すべての有効値の説明については、[87 ページ](#)の「フィールドオブジェクト」を参照)。
- send\_attribute

---

**注：** 環境テナントがICONアプリケーション オブジェクトに表示されない場合は、これを追加してから、テナント固有のオプションを設定します。

---

デフォルトでは、Configuration Managerにテナントの[付加情報]タブは表示されません。テナントの[付加情報]タブを表示するには、以下の手順を行います。

1. [ビュー] > [オプション] を選択します。
2. [オブジェクト プロパティで [付加情報] タブを表示] チェック ボックスを選択します。
3. [OK] をクリックします。





# 8

## ユーザ イベントとカスタム定義の状態の処理

Interaction Concentratorの顧客定義の状態のサポートは、従来のCall Concentratorとの互換性を維持するためのものです。この機能によって、カスタマイズされたデスクトップアプリケーションで、顧客が既存のシステムを活用できます。この章では、カスタム エージェント状態に関するデータとボイス コールに添付される共通データをInteraction Concentratorで使用方法について説明します。

この章には以下の項があります。

- [Interaction Concentrator のカスタム状態 \(91 ページ\)](#)
- [EventUserEvent に含まれるデータの格納 \(92 ページ\)](#)
- [カスタム状態の使用 \(92 ページ\)](#)

---

### Interaction Concentrator のカスタム状態

カスタム状態という用語は、エンドポイントについて顧客が定義した状態をいいます。適切に構成されている場合は、Interaction Concentrator (ICON) が T-Server の EventUserEvent に含まれるデータの処理を行い、以下のような関連付けを格納できます。

- 共通データとボイス コール (または該当する場合は通話パーティ)
- カスタム状態とボイス コール (または該当する場合は通話パーティ)

デバイスのアクティブな呼と直前の呼(カスタム状態の開始直前または EventUserEvent の到着直前に終了した呼)の両方が関連付けに属することができます。

## EventUserEvent に含まれるデータの格納

ICONはEventUserEventに含まれるデータと呼ぶユーザ データとは個別に処理します。

**共通データ** カスタム状態をサポートするように構成されている場合、ICONが EventUserEvent の共通データを2つのテーブルG\_CUSTOM\_DATA\_S と G\_CUSTOM\_DATA\_Pに格納します。これらはICONのインストールスクリプトによって作成されます。顧客定義のキーの値を格納するために、一意のキーを構成できます。添付データでは重複したキー名は使用できません。

**注：** Interaction Concentrator リリース 7.5 および 7.6 では、顧客が作成したテーブルまたは追加の顧客が作成したフィールドに共通データを格納することはできません。

**カスタム状態** ICONは、EventUserEventに含まれるカスタム状態に関するデータを G\_CUSTOM\_STATESテーブルに格納します。これは、ICONのインストールスクリプトによって作成されます。

カスタム状態が終了すると、ICONが情報をIDBに書き込みます。カスタム状態の滞留による問題を回避するために、エージェント ログインセッションが終了すると、ICONがアクティブなカスタム状態をすべて消去します。

カスタム データ テーブルおよびカスタム状態テーブルの詳細については、ご使用のRDBMSについて、『Interaction Concentrator 7.6 Physical Data Model』を参照してください。

## カスタム状態の使用

カスタム状態と共通データに関するレポートのために、ICON によって情報を格納するには、以下の手順を行う必要があります。

- 以下に示す適切な ICON アプリケーションの構成オプションを設定します。
  - ◆ AgentRecordUserTypes
  - ◆ AgentUserFields
  - ◆ EventData
  - ◆ GlobalData
  - ◆ store-event-data

AgentRecordUserTypes オプションによってカスタム エージェント状態を定義します。

- エージェント デスクトップ アプリケーションを構成して、適切な KVP (Key-Value Pairs : キーと値のペア) が T-Server に送信され、EventUserEvent の UserData に格納されるように設定します (「[エージェント デスクトップ アプリケーションの構成](#)」を参照)。

## エージェント デスクトップ アプリケーションの構成

T-Serverから受信するEventUserEventのUserDataに含まれる情報に基づいて、ICONがカスタム状態の開始と終了を記録します。エージェントデスクトップアプリケーションを構成して、EventUserEventのUserDataのために適切なKVP情報をT-Serverに送信できるようにする必要があります。

### カスタム状態の記録開始

ICONがカスタム状態の記録を開始するには、デスクトップアプリケーションから以下のKVPを送信する必要があります。

キー = "<StateKeyName>"、値 = "+"

#### 例

"AfterCallWork", "+"

### カスタムデータの送信

ICONがアクティブなカスタム状態に関する追加情報を格納するには、デスクトップアプリケーションから以下のKVPを送信する必要があります。

キー = "<CommentKey>"、値 = "<StateCode>, <Comment>"

同じカスタム状態に複数のコメントキーを指定できます。ただし、1つのコメントキーにICONが格納できる値は1つだけです。同じコメントキーに対して複数のKVPを送信すると、ICONが最後の値だけを格納します。

#### 例

"Comment", "207, This is data about the state"

"Explanation", "207, This is more data about the state"

"Explanation", "207, This is more, changed data about the state"

この例では、ICONが以下の値を格納します。

- Comment フィールドの状態 207 の場合 : This is data about the state
- Explanation フィールドの状態 207 の場合 : This is more, changed data about the state

### カスタム状態の記録停止

ICONがカスタム状態の記録を停止するには、デスクトップアプリケーションから以下のKVPを送信する必要があります。

キー = "<StateKeyName>"、値 = "-"

#### 例

"AfterCallWork", "-"

### 複数のカスタム状態の同時使用

各タイプのカスタム状態に対して一度にアクティブにできるのは、1つのDNの1つの状態だけです。ただし、ICONは複数の異なる状態を個別に同時処理することができます。たとえば、1つのDNの2つの異なる状態をアクティブにし、それぞれが異なるデータを持つことができます。ICONは添付データの重複したキー名をサポートしないため、同じキー名のKVPを1つのEventUserEventで送信することはできません(「例」を参照)。

**例**

```
"AfterCallWork", "+"  
"Break", "+"  
"Comment", "207, This is data about the call"  
"Comment", "208, This is data about the break"  
"Break", "-"  
"AfterCallWork", "-"
```

---

**注：** 上記の例で、ICON がキー値を格納するのは、カスタム状態 “207, This is data about the call” のみです。

---



パート



## パート II : Interaction Concentrator の管理

『Interaction Concentrator User's Guide』のパート II には、Interaction Concentrator の継続的な保守と管理を行うために必要な情報、および起動時とランタイムのトラブルシューティングについての情報が記載されています。また、Interaction Concentrator の機能に関する詳細な情報も記載されています。

パート I には以下の章があります。

- [97 ページ](#)の第 9 章「Interaction Concentrator の起動と停止」
- [107 ページ](#)の第 10 章「Interaction Concentrator のモニタ」
- [109 ページ](#)の第 11 章「IDB データのフィルタリング」
- [115 ページ](#)の第 12 章「Interaction Concentrator の HA の実装」
- [127 ページ](#)の第 13 章「構成の変更の再同期」
- [141 ページ](#)の第 14 章「特別なストアドプロシージャの使用」
- [163 ページ](#)の第 15 章「ICON システムと構築に関するトラブルシューティング」





# 9

## Interaction Concentrator の 起動と停止

この章では、Interaction Concentrator の起動の前提条件、および Interaction Concentrator (ICON) の起動と停止の手順について説明します。以下の項があります。

- [概要 \(97 ページ\)](#)
- [コマンドラインパラメータ \(98 ページ\)](#)
- [ICON の起動 \(98 ページ\)](#)
- [ICON の停止 \(103 ページ\)](#)

---

### 概要

Interaction Concentrator コンポーネントを起動およびシャットダウンするには、Management Layer、起動ファイル、手動の手順、または Services Manager を使用します。

通常、これらの方法はすべて、サーバアプリケーションと実行可能ファイル名のコマンドラインパラメータを必要とします。次の項では、ほとんどの Genesys サーバアプリケーションに共通のコマンドラインパラメータについて説明します。その後の項では、起動とシャットダウンの手順について説明します。

---

**注：** 起動のために Management Layer、起動ファイル、および Services Manager を使用する詳細については、『*Framework 7.6 Deployment Guide*』を参照してください。

---

## コマンドラインパラメータ

Interaction Concentrator では、以下の起動コマンドラインパラメータがサポートされます。

-host	Configuration Server が実行されるホストの名前。
-port	Configuration Server に接続するためにクライアントアプリケーションが使用する通信ポート。
-app	Configuration Database で構成されたとおりの正確なアプリケーションの名前。
-v	コンポーネントのバージョン。このパラメータを指定してもアプリケーションは起動せずに、バージョン番号が返されます。大文字 (V) または小文字 (v) を使用できます。
-lmspath	アプリケーションがログ イベントを生成するために使用するログメッセージファイルの完全なパス (common.lms という共通ファイルおよび *.lms 拡張子が付いたアプリケーション固有のファイル)。共通ファイルとアプリケーション固有のログメッセージファイルが、アプリケーションの作業ディレクトリ以外のディレクトリに置かれている場合に、このパラメータを使用します。たとえば、アプリケーションの作業ディレクトリが、アプリケーションが最初にインストールされたディレクトリと異なる場合などです。起動コマンドラインで実行可能ファイルの完全なパス (たとえば、c:\gcti\multiserver.exe) を指定した場合、実行可能ファイルとして指定したパスが *.lms ファイルを見つけるために使用され、lmspath パラメータの値は無視されます。

**警告!** 起動時に \*.lms ファイルを検索しないアプリケーションの場合は、アプリケーション固有のログ イベントを生成して、Message Server に送信することはできません。

## ICON の起動

この項では、ICON サーバの手動の起動手順について説明します。以下のいずれかの方法で ICON を起動できます。

- SCI から (99 ページを参照)
- UNIX で手動で (100 ページを参照)
- Windows で手動で (101 ページを参照)
- Windows で Windows のサービスとして (102 ページを参照)

## Solution Control Interface を使用した ICON の起動

Solution Control Interface (SCI)を使用してICONを起動するには、以下の手順を行います。

---

### 手順： SCIによるICONの起動

#### 前提条件

ICONを起動する前に、以下のアプリケーションを実行することをお奨めします。

- IDB へのアクセスを提供する DB Server
- 関係型データベース管理システム
- T-Server
- Outbound Contact Server (ICON が OCS からデータを収集するように構成されている場合)
- Interaction Server (ICONがMultimediaからデータを収集するように構成されている場合)

ICONが添付データを格納するように構成されている場合は、ICONの作業ディレクトリに適切な添付データ仕様ファイルが置かれていることを確認してください(デフォルトでは、ICONがccon\_adata\_spec.xmlファイルを使用)。

起動後または再起動後に、短時間、ICONがログに [cp:...] またはFSMエラーを記録する場合があります。これらのエラーは、ICONが起動または再起動されたときにインタラクションがすでに進行中であったため、ICONが解決できないインタラクションの要素に遭遇したことが原因で発生します。このエラーは無視して問題ありません。

Interaction Concentratorが依存するGenesysコンポーネントの起動手順の詳細については、以下のマニュアルを参照してください。

- 『*Framework 7.6 Deployment Guide*』
- ご使用の T-Server タイプの 『*Framework 7.6 T-Server Deployment Guide*』
- 『*Framework 7.6 DB Server User's Guide*』
- 『*Outbound Contact 7.6 Deployment Guide*』
- 『*Multimedia 7.6 Deployment Guide*』

#### 手順開始

1. SCI の [ アプリケーション ] ビューのリスト ペインで、ICON アプリケーション オブジェクトを選択します。
2. 以下のいずれかの手順を行います。
  - ツールバーで [ 開始 ] ボタンをクリックします。
  - [ アクション ] メニューから [ 開始 ] を選択します。

- アプリケーション オブジェクトを右クリックしてショートカット メニューを表示し、[ 開始 ] を選択します。
3. 確認ボックスが表示されたら、[ はい ] をクリックします。  
SCI が Interaction Concentrator アプリケーションを起動します。

### 手順終了

### 次のステップ

- これで、SCI を使用した ICON の起動が完了しました。

## 手動による ICON の起動

UNIXで手動でICONを起動するには、以下の手順を行います。

---

### 手順 :

### 手動による ICON の起動 (UNIX の場合 )

#### 手順開始

1. ICON をインストールしたディレクトリに移動します。
2. ICON 実行可能ファイルの名前を入力し、その後に適切なコマンドラインパラメータを付けます。以下に構文を示します。

```
./icon -host <hostname> -port <portno> -app <application>
```

ここで、

- *hostname*。Configuration Server が実行されているホストの名前
- *portno*。クライアントアプリケーションが Configuration Server に接続するために使用する通信ポート
- *application*。Configuration Server に対して定義された Interaction Concentrator アプリケーション オブジェクトの名前

---

**注:** ホスト名またはアプリケーション名にスペースやハイフン (-) が含まれる場合は、二重引用符で囲みます。

たとえば、ホストが `cs-host`、ポートが `2020`、名前が `ICON 03` の場合に、コマンドラインパラメータで `ICON` を起動するには、以下のように入力します。

```
./icon -host "cs-host" -port 2020 -app "ICON 03"
```

---

### 手順終了

## 次のステップ

- これで、UNIX で手動で ICON を起動するためのすべてのステップが完了しました。

## ICON の起動 (Windows の場合 )

Windows で ICON を起動するには、以下の手順を行います。

---

### 手順 :

### ICON の起動 (Windows の場合 )

**目的 :** [ スタート ] > [ プログラム ] メニューまたはコンソール ウィンドウを使用して ICON を起動します。

#### 手順開始

1. コンソール ウィンドウを開きます。
2. Interaction Concentrator をインストールしたディレクトリに移動します。
3. 以下のコマンドラインを入力します。

```
icon.exe -host <hostname> -port <portno> -app <application>
```

ここで、

- *hostname*。Configuration Server が実行されているホストの名前
- *portno*。クライアントアプリケーションが Configuration Server に接続するために使用する通信ポート
- *application*。Configuration Server に対して定義された Interaction Concentrator アプリケーション オブジェクトの名前

---

**注 :** ホスト名またはアプリケーション名にスペースやハイフン (-) が含まれる場合は、二重引用符で囲みます。

たとえば、ホストが `cs-host`、ポートが `2020`、名前が `ICON 03` の場合に、コマンドライン パラメータで `ICON` を起動するには、以下のように入力します。

```
icon.exe -host "cs-host" -port 2020 -app "ICON 03"
```

---

#### 手順終了

## 次のステップ

- これで、Windows で ICON を起動するためのすべてのステップが完了しました。

## Windows のサービスとしての ICON の起動

Microsoft Windows プラットフォームの場合、デフォルトでは、インストールプロセスによって Interaction Concentrator が Windows のサービスとしてインストールされます。Windows のサービスとして実行される ICON を停止し、再度 Windows のサービスとして起動するには、以下の手順を行います。

---

### 手順 :

## Windows のサービスとしての ICON の起動

### 手順開始

1. Windows のコントロール パネルを開き、[ サービス ] アイコンをダブルクリックします。[ サービス ] ダイアログ ボックスが開きます。
2. [ サービス ] リスト ボックスで ICON サービスを選択し、[ 開始 ] をクリックします。Interaction Concentrator が Windows のサービスとして動作しないように指定した場合は、このアプリケーションの [ 開始 ] オプションは使用できません。

---

**注 :** ユーザ インタフェースを無効にして、Local Control Agent (LCA) を Windows のサービスとしてインストールできます。この場合、LCA と ICON の両方で [ デスクトップとの対話をサービスに許可 ] チェック ボックスを選択しない限り、SCI から起動されるすべてのサーバがコンソールなしで起動します。

---

### 手順終了

### 次のステップ

- これで、Windows のサービスとして ICON を起動するためのすべてのステップが完了しました。

---

## ICON の停止

以下のいずれかの方法でICONを停止できます。

- SCI から (103 ページを参照) (この方法を推奨)
- UNIX で手動で (104 ページを参照)
- Windows で手動で (105 ページを参照)
- Windows で Windows のサービスとして (105 ページを参照)

---

**注：** ICON が自動起動するのを防ぐために、Configuration Manager で ICON アプリケーション オブジェクトから、必ず autorestart プロパティを消去してください。

---

## Solution Control Interface を使用した ICON の停止

LCAとSCSを使用している場合は、以下の手順で、SCIでICONを停止します。

---

### 手順：

### SCI を使用した ICON の停止

#### 手順開始

1. SCI の [ アプリケーション ] ビューのリスト ペインで、ICON アプリケーション オブジェクトを選択します。
2. 以下のいずれかの手順を行います。
  - ツールバーで [ 停止 ] をクリックします。
  - [ アクション ] メニューから [ 停止 ] を選択します。
  - アプリケーション オブジェクトを右クリックしてショートカット メニューを表示し、[ 停止 ] を選択します。
3. 確認ボックスが表示されたら、[ はい ] をクリックします。  
SCI が Interaction Concentrator アプリケーションを停止します。

#### 手順終了

#### 次のステップ

- これで、SCI を使用した ICON の停止が完了しました。

## ICON の停止 (UNIX の場合)

以下のいずれかの手順で、UNIX で ICON を停止します。

---

### 手順 :

#### UNIX のコマンドラインを使用した ICON の停止

##### 手順開始

- コマンドラインに以下のコマンドを入力します。  
`kill -SIGTERM <processid>`  
<processid> はアプリケーションの UNIX プロセス ID です。

##### 手順終了

##### 次のステップ

- これで、コマンドラインから ICON を停止するステップが完了しました。

---

### 手順 :

#### UNIX のコンソール ウィンドウを使用した ICON の停止

##### 手順開始

- アクティブなコンソール ウィンドウで CTRL+C を押します。

##### 手順終了

##### 次のステップ

- これで、コンソール ウィンドウから ICON を停止するステップが完了しました。

---

**注 :** LCA と SCS を使用している場合は、SCI を使用して ICON を停止することもできます (103 ページの「Solution Control Interface を使用した ICON の停止」を参照)。

---

## ICON の停止 (Windows の場合 )

Windows のサービスではなくアプリケーションとして ICON を実行している場合は、以下の手順で ICON を停止します。

---

### 手順 :

### Windows のコンソール ウィンドウを使用した ICON の停止

#### 手順開始

- アプリケーションのコンソール ウィンドウで CTRL+C を押します。

#### 手順終了

#### 次のステップ

- これで、コンソール ウィンドウから ICON を停止するステップが完了しました。

---

**注 :** ICON を Windows のサービスとして実行している場合は、Services Control Manager から ICON を停止する必要があります (「[Windows のサービスとしての ICON の停止](#)」を参照)。

---

## Windows のサービスとしての ICON の停止

Windows のサービスとして実行されている Interaction Concentrator を停止するには、以下の手順を行います。

---

### 手順 :

### Windows のサービスとして実行される ICON の停止

#### 手順開始

1. コントロール パネルを開き、[ サービス ] アイコンをダブルクリックします。[ サービス ] ダイアログ ボックスが開きます。
2. [ サービス ] リスト ボックスで ICON サービスを選択し、[ 停止 ] をクリックします。

#### 手順終了

#### 次のステップ

- これで、Windows のサービスとして実行される ICON を停止するためのすべてのステップが完了しました。





# 10 Interaction Concentrator の モニタ

この章では、HTTP Listener にアクセスして Interaction Concentrator のパフォーマンスをモニタしてレポートする方法を説明します。以下の項があります。

- [パフォーマンス カウンタへのアクセス \(107 ページ\)](#)

---

## パフォーマンス カウンタへのアクセス

HTTP Listener を構成して、ブラウザからパフォーマンス カウンタ ページにアクセスできます。このパフォーマンス カウンタを使用して、ICON のパフォーマンスをリアルタイムでレポートし、パフォーマンスと使用のパターンを特定して、ICON の構成を細かく調整できます。必要に応じて、トラブルシューティングのためにパフォーマンス カウンタ ページにアクセスすることもできます。

---

**注：** パフォーマンス カウンタ ページにアクセスすると、ICON のパフォーマンスに大きく影響します。このため、これらのページに定期的にアクセスすることはお奨めできません。

---

HTTP Listener を構成するには、HTTP 待機ポートが記述されている ICON アプリケーション オブジェクトに listeners セクションを作成します。HTTP Listener を構成したら、以下の手順で、ICON の実行中にパフォーマンス カウンタ ページにアクセスできます。

- ブラウザ ウィンドウを開いて、以下のアドレスに移動します。

`http://<host>:<port>`

ここで、

- ◆ host は、ICON サーバのホスト名
- ◆ port は、ICON アプリケーション オブジェクトの [オプション] タブにある <http-connection> セクションの port オプションに対して設定した値

HTTP Listener の構成の詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のオプションの構成についての章を参照してください。

### パフォーマンス カウンタ ページ

パフォーマンス カウンタ ページには、以下のタイプの情報が表示されます。

- 構成オブジェクトの統計—たとえば、ICON が読み取る DN、構成員、作業場所、およびエージェント ログイン オブジェクトの数
  - CTI の統計—たとえば、アクティブな呼、パーティ、エージェントセッションに関する詳細情報、アクティブな T-Server 接続に関する情報、CTI イベントに関する統計 (呼処理エラーの数、呼とイベントの平均ボリューム)、およびエージェント ログインセッションに関する累積統計
- マルチメディア インタラクションについても同様の情報が提供されます。
- ICON の統計—呼とエージェントセッションなど、主な ICON エンティティに関連する現在のアクティビティについてのメモリ割り当てに関する情報
  - データベース書き込みの統計—ICON インスタンス全体および DAP 役割別のデータベース書き込みアクティビティに関する情報と統計

図 9 は、サンプルのパフォーマンス カウンタ ページ (メイン Database Writer ページ) を示しています。

**注：** 図 9 は Web ページの例です。実際の画面はこれとは大幅に異なる場合があります。

Database Writer	
Status	Normal
Status time	2007-01-12 12:34:53
Error message	
Output queue backlog	0
Total data records written	50000
Total data records written/sec	0
Total records queued	50000
Records queued from calculation time	0
Records queued in previous 15 minutes	0
Last calculation time	2007-01-12 13:04:54
Components	
<a href="#">Configuration History</a>	
<a href="#">Call/party</a>	
<a href="#">User data</a>	
<a href="#">Agent login/session</a>	
<a href="#">Outbound</a>	

図 9: メイン Database Writer ページ



# 11

## IDB データのフィルタリング

この章では、Interaction Concentrator 7.6のデータ フィルタリング機能と、ICON アプリケーション オブジェクトでオプションを設定してこの機能を構成する方法について説明します。

データ フィルタリング機能を使用して、IDBに格納しないデータを指定してIDBのサイズを小さくし、データベースのパフォーマンスを向上させることができます。

この章には以下の項があります。

- [概要 \(109 ページ\)](#)
- [フィルタリング可能なデータ \(110 ページ\)](#)

---

### 概要

ICON のデータ フィルタリング機能で、レポートの要件に合わせて、IDB に格納するデータのタイプを制御できます。特定のデータのタイプを IDB に格納するものから除外することによって、以下のようなメリットがあります。

- データベース領域の節約 (IDB のサイズを小さくする)
- IDB のパフォーマンス向上
- 下流のレポート アプリケーションのパフォーマンス向上 (Genesys Info Mart など)

この機能を実装するには、Interaction Concentrator アプリケーション オブジェクトの filter-data セクションで新しい構成オプションを設定します。オプションを設定することにより、対応する IDB テーブルに ICON がデータを書き込まなくなるため、オプションを設定する前に以降の項を参照して、各タイプのデータがレポートに必要なかどうか慎重に検討してください。IDB スキーマとテーブルの概要については、[27 ページ](#)の第2章「IDB スキーマについて」を参照してください。

---

**注：** デフォルトでは、フィルタリングの構成オプションは `false` に設定されており、入手可能なすべてのデータが IDB に格納されます。

---

## フィルタリング可能なデータ

この項で取り上げるデータはフィルタリング可能です。ICONアプリケーションオブジェクトでさまざまな構成オプションを設定して、IDBの格納対象に含めたり除外したりできます。

---

**注：** この項に記載された情報に加えて、ICON で構成オプションを設定する詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』を参照してください。前述の IDB テーブルに格納されるデータの詳細については、ご使用の RDBMS タイプについて、『*Interaction Concentrator 7.6 Physical Data Model*』を参照してください。

---

## パーティ履歴データ

G\_PARTY\_HISTORYテーブルには、パーティ状態の履歴が変化した時期に関する通話パーティ情報が格納されます。配信デバイスの場合のみ、デバイスへのキューイングおよびデバイスのみからの配信に関する情報をこのテーブルに格納できます(たとえば、配信デバイスは呼び出し状態または保留状態を持つことができない)。

SIPおよび音声インタラクションの場合のみ、ACDキュー、ルーティングポイント、仮想ルーティングポイントなどの配信デバイスに関するパーティ履歴情報をIDBへの格納対象から除外できます。ICONアプリケーションで`acd-party-history`構成オプションを`true`に設定すると、ICONがパーティ関連情報を G\_PARTY\_HISTORYテーブルに書き込みません。

## パーティメトリクス

ICONは、ACDキュー、ルーティングポイント、仮想ルーティングポイントなどの配信デバイスの事前に計算されたパーティメトリクスを収集し、この情報をG\_PARTY\_STATテーブルに格納します。

SIPおよび音声インタラクションの場合のみ、配信デバイスのパーティメトリクスデータをIDBへの格納対象から除外できます。除外するには、ICONアプリケーションオブジェクトで、`acd-party-metrics`構成オプションを`true`に設定します。

## 外部パーティ

ICON は外部パーティ (たとえば、所定の交換機ドメイン外のインタラクション参加者) についての情報を収集して、以下の IDB テーブルに格納します。

- G\_PARTY
- G\_PARTY\_HISTORY
- G\_PARTY\_STAT

外部パーティ データを IDB に格納しない場合は、ICON アプリケーション オブジェクトで、external-party 構成オプションを true に設定します。

## ユーザ データ履歴

ICON が特定のキーの UserData 値の履歴全体を格納するように構成されている場合は、ICON が該当するキーの値の変化すべてについてデータを格納し、インタラクションの終了時にこの情報を以下の IDB テーブルに格納します。

- G\_USERDATA\_HISTORY
- G\_SECURE\_USERDATA\_HISTORY

ICON アプリケーション オブジェクトで、udata-history-terminated 構成オプションを true に設定すると、ICON は UserData キーの呼の終了時の値を IDB に格納しません。ただし、ICON は、呼の存続期間中に変化 (キーと値のペアの作成、追加、削除) が発生するたびに、UserData の変化の履歴をこれらのテーブルに格納します。表 15 に、これらの IDB テーブルに格納される情報を示します。

表 15: IDB に格納される変更情報のタイプ

値	列の変更タイプ	説明
1	created	1 は、呼が作成された瞬間に、キーの値が呼に添付されたことを示す。チャット インタラクションの場合は、eventInteractionSubmitted イベント (isOnline 属性が true) を受け取ったときに、ICON が変更タイプ 1 を割り当てる。
2	added	2 は、キーの値がたった今追加されたことを示す。
3	updated	3 は、キーの値が変化したことを示す。チャット インタラクションの場合は、eventPropertiesChanged イベント (isOnline 属性が true または false) を受け取ったときに、ICON が変更タイプ 3 を割り当てる。
4	deleted	4 は、UserData からキーが削除されたことを示す。
5	terminated	5 は、呼の終了時に ICON がキーの値を記録したことを示す。

## 呼のメトリクス

ICON は呼のメトリクスに関する情報を G\_CALL\_STAT テーブルに格納します。呼のメトリクスを IDB への格納対象から除外するには、ICON アプリケーション オブジェクトで、call-metrics 構成オプションを true に設定します。

## 呼の履歴

ICONは呼の履歴情報をG\_CALL\_HISTORYテーブルに格納します。呼の履歴情報をIDBへの格納対象から除外するには、ICONアプリケーション オブジェクトで、call-history構成オプションをtrueに設定します。

## インタラクション レコードの履歴

ICONはインタラクションレコードの履歴を収集して、この情報をG\_IR\_HISTORYテーブルに格納します。インタラクションレコードの履歴に関するデータをIDBへの格納対象から除外するには、ICONアプリケーション オブジェクトで、ir-history構成オプションをtrueに設定します。

## エージェント アクティビティ データ

ICON は、ログインセッションやエージェント状態などのエージェント アクティビティに関する情報を収集します。特定のタイプのデータを除外するように設定した場合 ( 以下のサブセクションを参照 ) 以外は、ICON がこの情報を以下の IDB テーブルに格納します。

- G\_LOGIN\_SESSION
- GX\_SESSION\_ENDPOINT
- G\_AGENT\_STATE\_HISTORY
- G\_AGENT\_STATE\_RC
- G\_DND\_HISTORY
- GS\_AGENT\_STAT
- GS\_AGENT\_STAT\_WM

エージェント アクティビティ情報をIDBに格納しない場合は、ICONアプリケーション オブジェクトで、gls-all構成オプションをtrueに設定します。ただし、各エージェントのIDはG\_PARTYテーブルに書き込まれます。

## エージェント メトリクス

ICON はエージェント状態の情報を収集して、以下の IDB テーブルに格納します。

- GS\_AGENT\_STAT
- GS\_AGENT\_STAT\_WM

エージェント状態の情報を除外するには、ICONアプリケーション オブジェクトで、gls-metrics構成オプションをtrueに設定します。

## IVR デバイスからのエージェント アクティビティ

IVR エンドポイントからエージェント ログインセッションが開始したときに、ICON がエージェント アクティビティに関するデータを収集し、この情報を以下の IDB テーブルに格納します。

- G\_LOGIN\_SESSION
- GX\_SESSION\_ENDPOINT
- G\_AGENT\_STATE\_HISTORY
- G\_AGENT\_STATE\_RC
- G\_DND\_HISTORY
- GS\_AGENT\_STAT
- GS\_AGENT\_STAT\_WM

IVR エンドポイントのエージェント アクティビティに関するデータを IDB への格納対象から除外するには、ICON アプリケーション オブジェクトで、`gls-ivr` 構成オプションを `true` に設定します。エージェントがログインした DN が IVR デバイスかどうかを ICON が検証し、IVR デバイスの場合は、このエージェントのアクティビティに関する情報は IDB に格納しません。さらに、IVR デバイスに関連付けられたパーティの場合は、ICON がエージェントの ID を `G_PARTY` テーブルに記録しません。

## 構成されていない構成員のエージェント アクティビティ

ICON はすべてのエージェント アクティビティに関するデータを収集して、この情報を IDB テーブルに格納します(112 ページの「エージェント アクティビティ データ」を参照)。ログイン ID が構成員構成オブジェクトに関連付けられていないエージェントのエージェント アクティビティに関するデータを、IDB への格納対象から除外することができます。除外するには、ICON アプリケーション オブジェクトで、`gls-no-person` 構成オプションを `true` に設定します。この場合、エージェント状態に関するイベントでレポートされた LoginID が、Configuration Database で構成された構成員オブジェクトに割り当てられているかどうかを ICON が検証します。割り当てられていない場合は、このエージェントのアクティビティに関する情報をテーブルに格納しません。

## エージェントの作業モード

ICON はエージェントの作業モードとエージェントの作業モードの変化に関するデータを収集して、以下の IDB テーブルに格納します。

- G\_AGENT\_STATE\_HISTORY
- G\_AGENT\_STATE\_RC
- GS\_AGENT\_STAT\_WM

エージェント状態の変化と一致しないエージェントの作業モードの変化に関するデータを除外することができます。除外するには、ICON アプリケーション オブジェクトで、`gls-wm` 構成オプションを `true` に設定します。この場合、ICON が `unknown` という値を IDB テーブルに記録します。

---

**注：** このオプションは、ICON の後処理の追跡機能には影響しません。

---

## エージェント キュー

ICON はエージェント キューに関する情報を収集して、以下の IDB テーブルに格納します。

- G\_AGENT\_STATE\_HISTORY
- G\_AGENT\_STATE\_RC
- GS\_AGENT\_STAT
- GS\_AGENT\_STAT\_WM
- GX\_SESSION\_ENDPOINT

エージェントがログインしているキューについての情報を除外するには、ICONアプリケーション オブジェクトで、gls-queue構成オプションをtrueに設定します。除外すると、ICONが上記の最初の4つのテーブルにキュー関連のデータを書き込みません。ただし、GX\_SESSION\_ENDPOINTテーブルには、エージェントがログインしているキューに関する情報を引き続き書き込みます。

## Service Observer データ

ICON は呼の observer の役割を持つパーティに関連したデータを収集して、この情報を以下の IDB テーブルに書き込みます。

- G\_PARTY
- GS\_PARTY\_STAT

observerの役割を持つパーティに関するデータをIDBに格納しない場合は、ICONアプリケーション オブジェクトで、observer-party構成オプションをtrueに設定します。



# 12 Interaction Concentrator の HA の実装

この章では、Interaction Concentrator の HA (High Availability : 高可用性) の設計と実装について説明します。リアルタイム音声、エージェント固有、アウトバウンド、および仮想キュー インタラクションデータを、HA ペアからなる IDB で抽出する方法について取り上げます。さらに、サーバ切断に関連するデータの不整合についても取り上げます。最後に、HA 環境における ICON アプリケーションと IDB の構成上の考慮点を取り上げます。

---

**注：** この章は、ETL エンジンの開発経験がある方、または Genesys Info Mart を使用して IDB から抽出したデータに基づいてレポートを作成した経験がある方を対象としています。

---

この章には以下の項があります。

- [概要 \(116 ページ\)](#)
- [ICON の HA モデル \(116 ページ\)](#)
- [HA IDB ペアからのデータ抽出 \(117 ページ\)](#)
- [構成データの HA \(122 ページ\)](#)
- [エージェント固有データの HA \(122 ページ\)](#)
- [サーバの切断 \(123 ページ\)](#)
- [ICON サーバの障害 \(124 ページ\)](#)
- [構成上の考慮点 \(124 ページ\)](#)

## 概要

Interaction Concentrator を高可用性に構築して、音声インタラクション、添付データ、エージェント状態、および仮想キューの詳細に関するレポートデータの冗長性を確保することができます。

HA 構成では、コンポーネントやネットワークの障害により、いずれかの ICON プロセスが音声インタラクション、添付データ、または仮想キュー データを IDB に格納できなくなっても、別の ICON プロセスがこれらのインタラクションを IDB に格納できる限り、データが失われることはありません。

さらに、Genesys Info Mart などの下流のレポート アプリケーションが HA ペアの少なくとも一方の IDB に接続できる限り、音声インタラクション、添付データ、仮想キュー データの抽出、変換、ロードを続行できます。

## ICON の HA モデル

Interaction Concentrator で使用される高可用性モデルは、ほとんどの Genesys サーバに実装される Genesys 標準の HA モデルとは大きく異なります。

### Genesys 標準 HA モデルとの違い

通常の Genesys サーバ (T-Server など) とは異なり、冗長 (HA) ペアとして構築される 2 つの Interaction Concentrator はプライマリ モードやバックアップ モードでは動作せず、さらに一方のサーバに障害が発生したときに、モードがバックアップからプライマリに切り替わることもありません。HA ペアの 2 つの Interaction Concentrator がスタンドアロンサーバとして並列で動作し、着信するデータを独立して処理します。

冗長ペアの Interaction Concentrator が同じ T-Server (スタンドアロン T-Server または冗長ペアのプライマリ T-Server) からイベントを受け取り、2 つの独立した IDB にデータを格納します。下流のレポート アプリケーションが両方の IDB から取得する冗長データの重複を解決します。図 10 に、ICON HA ソリューションの主なコンポーネントを示します。

**注：** 簡潔にするために、図 10 では Configuration Server と DB Server が省略されています。

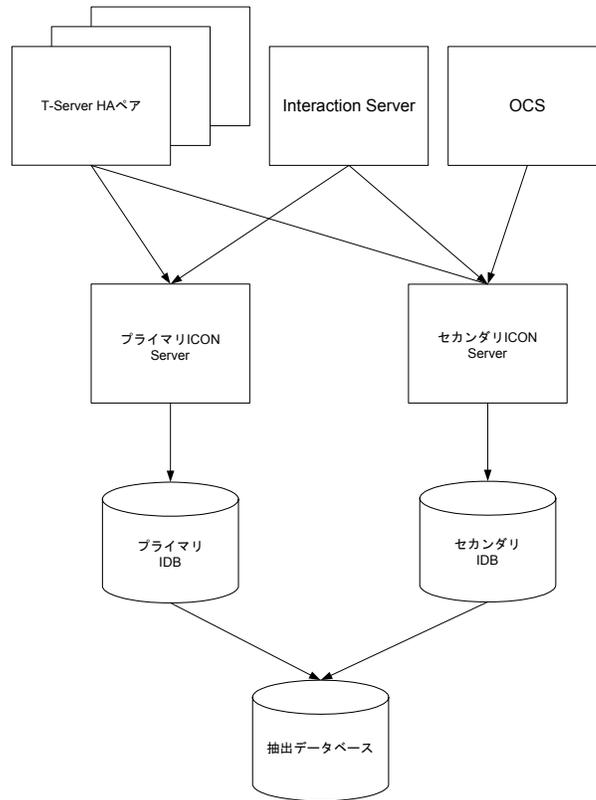


図 10: ICON の HA モデル

## HA IDB ペアからのデータ抽出

この項では、HAペアのIDBからデータを抽出する方法をいくつか取り上げます。推奨事項は、データのタイプと環境の複雑さに対応するデータ抽出方法を選択することを前提としています。

### リアルタイム インタラクション データ

音声固有インタラクションなどのリアルタイム インタラクション データは、以下の IDB テーブルに格納されます。

G_IR	G_CALL	G_CALL_USERDATA
G_IR_HISTORY	G_CALL_HISTORY	G_CALL-USERDATA-CUST
G_IS_LINK	G_CALL_STAT	G_CALL_USERDATA_CUST1
G_IS_LINK_HISTORY	G_PARTY	G_USERDATA_HISTORY
G_ROUTE_RESULT	G_PARTY_STAT	G_SECURE_USERDATA_HISTORY

## 抽出手順

プライマリおよびバックアップIDBの両方からG\_IRレコードを完全に抽出する抽出手順が必要です。

- 完全なレコードが G\_IR テーブルで MERGESTATE 3 に設定される。
- 増分抽出の場合は、ICON により GSYS\_MSEQ フィールドに入力される。

データが不完全な場合は、Interaction Concentratorがアイドル呼の解決手順を使用して、使用可能なデータを処理する決定をします。この手順はタイムアウトメカニズムに基づいており、一部のデータが失われてもデータ処理を続行できます。IDB内で、Interaction Concentratorが不完全だと検出したレコードに信頼性が低いものとしてマークを付けます。このようなレコードの詳細については、ご使用のRDBMSについて、『*Interaction Concentrator 7.6 Physical Data Model*』のG\_IR.GSYS\_EXT\_VCH2フィールドの説明を参照してください。

**注：** 音声データ マージのための gsysIRMerge ストアドプロシージャの詳細については、141 ページの「マージストアドプロシージャ」を参照してください。

図 11 に、Interaction Concentrator のデータ抽出モデルを示します。

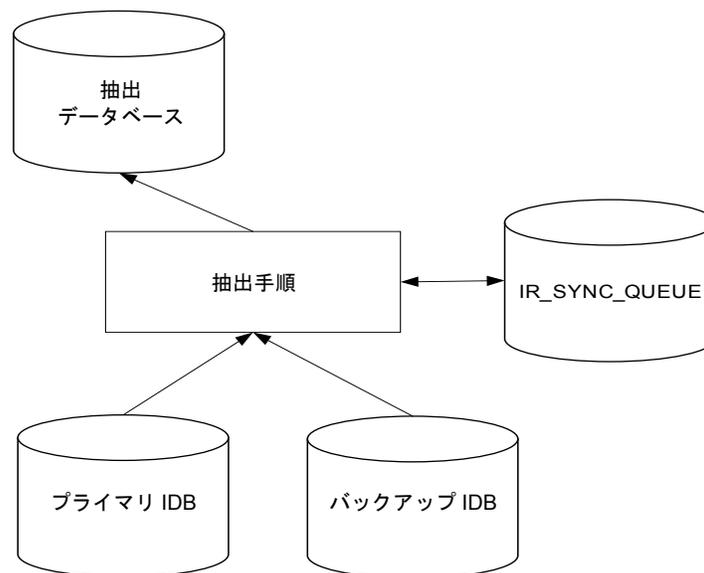


図 11: IDB のデータ抽出 — 論理モデル

IDB インスタンスから抽出されるレコードは、以下のシナリオのように処理されます。

**シナリオ 1:** IR\_SYNC\_QUEUE にレコードが見つからない場合

- レコードの終了時刻が指定されたタイムアウトより遅い場合：
  - ◆ レコードが無視される、または

- ◆ 抽出データベースに対してレコードの存在が検証される。レコードが存在しない場合は、終了時刻が指定のタイムアウトより早いレコードとして処理される。
- レコードの終了時刻が指定されたタイムアウトより早い場合：
  - ◆ レコードが抽出データベースに抽出される、かつ
  - ◆ レコード ID (IRID) が IR\_SYNC\_QUEUE に追加される (表 16 を参照)。

**シナリオ 2:** IR\_SYNC\_QUEUE にレコードが見つかった場合

- レコードが削除される。

**シナリオ 3:** IR\_SYNC\_TIMEOUT の期限が切れた場合

- IR キャッシュ保守手順によって、IR レコードがキャッシュから削除される。

**注:** IR\_SYNC\_TIMEOUT で指定する時間は、セカンダリ データベースから入手可能なデータの最大遅延より長くする必要があります。

表 16: IR\_SYNC\_QUEUE テーブルの構造

フィールド	型	説明
ID	numeric	データベースが生成するプライマリキー。
IRID	varchar (50)	G_IR 抽出レコードの IRID フィールドの値。
Extracted	datetime	IR レコードが抽出されたグリニッジ平均時の日付と時刻。IR キャッシュ保守手順がこの値を使用する。

## 前提条件

HA ペアの IDB からのレコード抽出は、以下の前提条件に基づいています。

- インタラクション関連のすべてのデータが IDB に格納される。
- 添付データはインタラクションデータと同じ IDB インスタンスに格納される。
- サーバの切断や接続障害に関連するデータの不整合は IDB が処理する。

## レコードの処理

Interaction Concentrator は、以下のテーブルのレコードを記載された順序で処理します。

1. G_IR_HISTORY	2. G_CALL
3. G_CALL_HISTORY	4. G_CALL_STAT
5. G_PARTY	6. G_PARTY_HISTORY
7. G_PARTY_STAT	8. G_IS_LINK
9. G_IS_LINK_HISTORY	10. G_ROUTE_RESULT
11. G_CALL_USERDATA	12. G_CALL_USERDATA_CUST
13. G_CALL_USERDATA_CUST1	14. G_CALL_USERDATA_CUST2
15. G_CALL_USERDATA_HISTORY	16. G_SECURE_USERDATA_HITSTORY

G\_IRテーブルのレコードは、データ抽出の基準レコードとして使用されます。IRIDフィールドの値をIDとして使用して、G\_IR\_HISTORYおよびG\_CALLテーブルから抽出するレコードを決定します。

G\_CALLテーブルからデータを抽出したら、CALLIDフィールドの値をIDとして使用して、以下のテーブルから抽出するレコードを決定します。

G_CALL_HISTORY	G_CALL_USERDATA_CUST
G_PARTY	G_CALL_USERDATA_CUST1
G_CALL_STAT	G_CALL_USERDATA_CUST2
G_USERDATA_HISTORY	G_IS_LINK
G_SECURE_USERDATA_HISTORY	G_IS_LINK_HISTORY

G\_PARTYからデータを抽出したら、PARTYIDフィールドの値をIDとして使用して、G\_PARTY\_HISTORYおよびG\_PARTY\_STATテーブルのレコードを決定します。

## 滞留レコード

ICON がシャットダウンしている間に以下のような状況が発生すると、ICON の再起動時にレコードが滞留する場合があります。

- エージェントセッションデータが変化する (たとえば、エージェントがログインまたはログアウトする)。
- アウトバウンドキャンペーン処理が完了する。
- 仮想キューから呼が送信される。
- 呼またはパーティが削除される。

T-Server側で滞留した呼やパーティによって、アイドル呼またはアイドルパーティが発生することもあります。

---

**注：** ICON が T-Server から切断しただけでは、アイドル呼は発生しません。ICON がアクティブな呼のスナップショットを取得して、それをメモリの呼と比較します。

---

## 仮想キュー固有のデータ

仮想キュー レコードは、両方のIDB インスタンスで一意のID を持ちます (G\_VIRTUAL\_QUEUE テーブルのVQID フィールド)。仮想キュー データの抽出手順は、リアルタイム インタラクション データと同様です。仮想キュー レコードは関連付けられた状況を持ち、これをデータ検証に使用できます。たとえば、Status フィールドに stuck (整数値 = 2) という値を持つ仮想キュー レコードを無視し、normal または abandoned の状況を持つ仮想キュー レコードを優先することができます。

## アウトバウンド データ

アウトバウンド データを抽出する手順は、音声データ インタラクションの場合と同様です。GO\_CAMPAIGN および GO\_CHAIN テーブルのレコードを基準レコードとして使用します。この2つのテーブルのレコードは、個別に抽出できます。

---

**注：** 終了したレコードだけが抽出対象となります。

---

アウトバウンド データの抽出手順では、さらに、以下のフィールドの値の組み合わせが一意のレコードID として使用されます。

SessID	OCSID
CallingListID	RECORDHANDLE
CHAINGUID	TYPE
CALLATTID	ADDED_TS

## キャンペーン レベル データの抽出

GO\_CAMPAIGN テーブルから終了したレコードを抽出する際には、SessID フィールドの値が一意のID として使用されます (SessID フィールドの同じ値が両方のIDB インスタンスに存在する)。この値を使用して、同じIDB インスタンスのGO\_CAMPAIGNHISTORY およびGO\_CAMPPROP\_HIST テーブルからデータを抽出します。

## チェーン レベル データの抽出

GO\_CHAIN テーブルから終了したレコードのデータを抽出する際には、CHAINGUID フィールドの値を一意のID として使用します (両方のIDB インスタンスに同じ値が存在する)。CHAINGUID フィールドの値を使用して、同じIDB インスタンスのGO\_CHAINREC\_HIST、GO\_RECORD、GO\_CUSTOM\_FIELDS、GO\_FIELDHIST、GO\_SECURE\_FIELDS、GO\_SEC\_FIELDHIST およびGO\_CHAIN\_CALL テーブルからデータを抽出します。

## 事前に計算されたメトリクスの抽出

OCS が一部のメトリクスを事前に計算して、このデータを Interaction Concentrator に送信します。OCS はこれらのメトリクスの一意の ID は提供しませんが、(メトリクスの計算対象の)オブジェクトの ID とメトリクスのタイムスタンプは、両方の IDB インスタンスで同一です。

---

## 構成データの HA

IDB に格納された構成オブジェクトを識別するために、Interaction Concentrator は Configuration Server が割り当てたデータベース ID (DBID) を使用します。この方法では、2 つの HA IDB に格納される所定の構成オブジェクトの DBID 値が同一の必要があります。さらに、ICON 7.6 では、GC\_\* および GCX\_\* テーブルの GSYS\_EXT\_INT1 フィールドで、構成データに信頼性のフラグが付きます。2 つの HA IDB の間で、同じ構成レコードの信頼性の値が異なる場合は、より信頼性の高いレコードからデータが抽出されます。詳細については、[131 ページ](#)の「信頼性フラグ」を参照してください。

---

## エージェント固有データの HA

ICON 7.6 は、ICON サーバの冗長ペアを使用することにより、エージェント固有データ (ログインセッションと後処理) の高可用性をサポートします。エージェントデータの高可用性をサポートするために、プライマリ T-Server が最初のエージェント ログインとタイムスタンプから AgentSessionID を作成します。AgentSessionID はバックアップ T-Server には送信されません。代わりに、(スイッチオーバー後に) バックアップ T-Server がプライマリになると、新しい AgentSessionID が認識されたすべてのエージェントセッションに割り当てられます。ICON はプライマリ T-Server 7.6 から受信した AgentSessionID だけを処理します。

ICON 7.6 が、まず、ログインセッションに関するデータを apstorage.db という持続的キャッシュファイルに格納します。これは、重複したログインセッションが IDB に格納されることを防止し、ログインセッションの滞留を防ぐためです。通知された AgentSessionID が IDB または ICON メモリに存在する場合は、ログインセッションが重複しているとみなされます。その他の場合は、既存のログインセッションに closed のマークを付け (たとえば、理由が stuck)、新規セッションを作成します。ICON は HA のエージェントデータを G\_LOGIN\_SESSION IDB テーブルに格納します。また、一意のエージェント固有データ (後処理など)、およびログインセッションの信頼性に関する情報と復元されたエージェント状態に関する情報を IDB に格納します。

---

**注：** agent-pstorage-name 構成オプションを使用して、apstorage.db という持続的キャッシュファイルの名前を変更できます。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』の構成オプションについての章を参照してください。

---

## サーバの切断

この項では、ICON HA 構成におけるサーバ切断の影響について説明します。サーバ切断とデータの不整合の関連についても説明します。

### T-Server の切断

ICON と T-Server との接続が失われると、IDB で以下のようなデータの不整合が発生する可能性があります。

- ICON 側から見た誤ったパーティ移行の結果、移行が失われる、またはパーティエラーレコードが生成される。
- 不明の時間に削除された呼。
- 以下の項目についての情報が失われる。
  - ◆ エージェント状態
  - ◆ 添付データ
  - ◆ 添付データの変更
  - ◆ パーティ
  - ◆ 呼のリンク

### CTI リンクの切断

CTI リンクの切断により、IDB で以下のようなデータの不整合が発生する可能性があります。

- アイドル呼
- 呼関連およびエージェント関連イベントの誤った移行
- 滞留パーティ ( 実際には関連付けられた呼インタラクションが削除されているのに、ICON がアクティブ ( 削除されていない ) としてレポートするパーティ )

### HA ペアのアクティブな T-Server の障害

HA ペアのアクティブな T-Server の障害により、IDB で以下のようなデータの不整合が発生する可能性があります。

- TEvents の誤った移行
- 滞留パーティ
- 失われた添付データ
- 不完全なイベントフローによるその他の不整合

## Interaction Server の切断

ICON が予期せず Interaction Server から切断することにより、IDB で以下のようなデータの不整合が発生する可能性があります。

- ICON 側から見た誤ったパーティ移行の結果、移行が失われる、またはパーティエラーレコードが生成される。
- 不明の時間に削除されたインタラクション。
- 滞留インタラクション
- 以下の項目についての情報が失われる。
  - ◆ エージェント状態
  - ◆ 添付データ

## OCS の切断

ICONが予期せずOutbound Contact Serverから切断することにより、IDBで以下のようなデータの不整合が発生する可能性があります。

- 以下の項目についての情報が失われる。
  - ◆ キャンペーン処理
  - ◆ コーリングリストの変更
  - ◆ OCS とボイス コール データのリンク
- OCS の事前計算されたメトリクスが失われる。

---

## ICON サーバの障害

ICON サーバの障害により、特定の ICON インスタンスのデータが失われたり、持続的キューに格納されたデータの復元に問題が発生する場合があります。その結果、IDB で以下のようなデータの不整合が発生する可能性があります。

- アイドル呼
- 滞留パーティ
- 失われたレコード

---

## 構成上の考慮点

HA の構築では、以下の点を考慮する必要があります。

- 両方の Interaction Concentrator アプリケーション オブジェクトで、構成オプションを正確に同一に設定する必要があります。これは Genesys の標準的な冗長ペアではないため、Configuration Server で 2 つの Interaction Concentrator の構成オプションを自動的に同期することはできません。

たとえば、冗長 ICON アプリケーションを構成して、音声インタラクションデータをペアの HA IDB に格納するには、以下のように設定します。

- ◆ 両方の ICON アプリケーション オブジェクトで、role オプションに gcc と gud が含まれるように設定します。これによって、両方の ICON アプリケーションが呼関連データと添付データを格納できます。
- ◆ これらの役割によって設定されるデータに影響する構成オプションについては、両方の ICON アプリケーションで同じオプション値を設定します。たとえば、仮想キューのモニタ、添付データの格納などについては、2 つのアプリケーションが同一の ICON 構成オプションを使用する必要があります。

構成オプションの設定の詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』を参照してください。

- 両方の Interaction Concentrator アプリケーション オブジェクトで、同じ T-Server への接続を設定する必要があります。
- 2 つの同一の IDB を作成する必要があります。同じ RDBMS タイプと同じバージョン番号を持ち、異なるホストに置かれた 2 つのデータベースを使用して、HA ペアの IDB をホストすることをお奨めします。
- それぞれの ICON が IDB に接続するための DAP (Database Access Point : データベース アクセス ポイント) を構成する必要があります。

Configuration Manager でアプリケーションと接続を構成する詳細については、『*Framework 7.6 Deployment Guide*』を参照してください。





# 13

## 構成の変更の再同期

適切に構成されている場合、ICONは最初に起動されたときに、Configuration Serverから構成オブジェクトに関する情報を収集します。動作中にICONは、Configuration Serverからの動的なリアルタイム通知をモニタして、これらのオブジェクトに加えられる変更を追跡します。ICONは、新規オブジェクトの追加と既存のオブジェクトの削除や更新についてのConfiguration Server情報を、IDBにあるGC\_というプレフィックスが付いたテーブルに格納します。また、オブジェクトの関係についてのConfiguration Server情報は、GCX\_が付いたIDBテーブルに格納します。

特定の条件下では、IDBデータをConfiguration Serverデータと同期できない場合があります。このような場合、動作中のICONとICONデータに依存する下流のアプリケーションが、解釈が困難な状況に陥る可能性があります。こうした場合には、IDBと現在の構成データを再同期したいと考えるでしょう。

---

**注：** 構成の変更を格納するように ICON を構成した場合は、ICON が ( 中断することなく ) 動作を継続することが重要です。これによって、IDB と Configuration Server データの同期が失われる可能性が小さくなります。

---

この章では、IDB データの強制再同期が必要になる条件、および IDB を再同期するために必要な手順について説明します。また、万が一クラッシュが発生したときにデータを回復するために、Configuration Database を復元する方法も説明します。以下の項があります。

- [ICON が構成データを収集する方法 \(128 ページ\)](#)
- [再同期の動作 \(130 ページ\)](#)
- [再同期に必要な時間 \(132 ページ\)](#)
- [IDB の再同期が必要な状況 \(132 ページ\)](#)
- [構成データの再同期の方法 \(135 ページ\)](#)
- [再同期の推奨事項 \(136 ページ\)](#)
- [バックアップを使用した Configuration Database の復元 \(137 ページ\)](#)

## ICON が構成データを収集する方法

ICON の role 構成オプションの値によって、ICON が構成関連のデータを収集して格納するかどうかが決まります。このオプションに `cfg` という値がある場合は、ICON が以下のような場合にデータを収集します。

- 起動時 (ICON を初めて構築し、IDB が空の場合)
- 永続キューを使用できない場合
- リアルタイム オブジェクト変更通知があった場合
- Configuration Server の履歴ログ ファイルを受信した場合
- ユーザから再同期の要求があった場合

### 起動時の Configuration Database の読み取り

最初に起動したとき、またはローカル キャッシュ ファイルを使用できないとき (「永続キャッシュを使用できない場合」を参照) に、すべてのアクティブな構成オブジェクトおよびアクティブな関係について、ICON が Configuration Server に対してクエリを実行します。これらのオブジェクトと関係について収集した情報が格納されたローカル ファイル (永続キャッシュ) を ICON がロードします。このローカル永続キャッシュ ファイルの名前は `cfg-sync.db` です。次に、更新されたトランザクションを永続キューに送信し、IDB にある以下の構成関連テーブルを更新します。

- GCX\_AGENT\_PLACE
- GCX\_CAMPGROUP\_INFO
- GCX\_CAMPLIST\_INFO
- GCX\_ENDPOINT\_PLACE
- GCX\_FORMAT\_FIELD
- GCX\_GROUP\_AGENT
- GCX\_GROUP\_ENDPOINT
- GCX\_GROUP\_PLACE
- GCX\_GROUP\_ROUTEDN
- GCX\_LIST\_TREATMENT
- GCX\_LOGIN\_INFO
- GCX\_OBJTABLE\_RECORD
- GCX\_SKILL\_LEVEL
- GCX\_SUBCODE
- GC\_ACTION\_CODE
- GC\_AGENT
- GC\_APPLICATION
- GC\_ATTR\_VALUE
- GC\_BUS\_ATTRIBUTE
- GC\_CALLING\_LIST
- GC\_CAMPAIGN
- GC\_ENDPOINT
- GC\_FIELD
- GC\_FILTER
- GC\_FOLDER
- GC\_FORMAT
- GC\_GROUP
- GC\_IVR
- GC\_IVRPORT
- GC\_LOGIN
- GC\_OBJ\_TABLE
- GC\_PLACE
- GC\_SCRIPT
- GC\_SKILL
- GC\_SWITCH
- GC\_TABLE\_ACCESS
- GC\_TENANT
- GC\_TIME\_ZONE
- GC\_TREATMENT
- GC\_VOICE\_PROMPT

**注：** ICON が GC\_APPLICATION テーブルに情報を格納するのは、Outbound Contact Server、CPD Server、T-Server および Agent Desktop アプリケーション タイプの場合のみです。

## 永続キャッシュを使用できない場合

通常の起動時(IDBが空の初めての起動時以外)に永続キャッシュを使用できない場合は、ICONが自動的にIDBの再同期を実行します。ICONが、Configuration Serverから受信して最後に処理したリアルタイム通知を使用して、起動時にIDBの内容を検証します。永続キューを使用できないため、最後の通知についての情報がない場合は、ICONがConfiguration Serverの構成関連のすべての情報を要求し、永続キャッシュを回復します。

## ICON が動的通知を受信した場合

ICONはConfiguration Serverのクライアントです。両方のアプリケーションが動作し、Configuration Manager内で、構成オブジェクトまたは構成オブジェクトと他のオブジェクトとの関係に変更が加えられるたびに、Configuration Serverが即座に変更についてクライアントに通知します。Configuration Database内で直接オブジェクトが変更された場合は、Genesysはこのような通知をサポートしません。永続キャッシュは常にConfiguration Databaseと同期するように設計されています。ICONは通知を受け取ると、即座にその情報を永続メモリキャッシュに送信し、Configuration Serverでオブジェクトが変更されたときの実際のタイムスタンプを使用して、情報を適切なIDBテーブルに記録します。

## ICON が構成履歴ログを読み取る場合

Configuration Serverには履歴ログが保持されます。これによって、サービス中断によって終了したセッションをクライアントが復元し、中断中に構成オブジェクトに加えられた変更を要求することができます。構成オブジェクトに加えられた動的変更は、Configuration Serverが直接レポートします。Configuration Serverに接続するたびに、ICONがこの情報を要求します。

以前のリリースのConfiguration Server (7.6より前)の場合、Configuration Serverのhistory-log-activeオプションがtrueに設定されている必要があります。falseに設定されていると、構成の変更が履歴ログファイルに記録されません。このような場合に、ICONとConfiguration Serverの接続が失われると、切断中の構成変更についての情報を後で入手することができません。ただし、このオプションをfalseに設定しても、再同期は実行されます。

リリース 7.6 では、以下に示す Configuration Server 7.6 の [history-log] 構成オプションを設定して、履歴ログの機能を制御できます。

- all
- expiration
- client-expiration
- max-records
- active
- failsafe-store-processing-

---

**注：** `failsafe-store-processing` を `false` に設定すると、履歴ログ データベースが完全に保持されない場合があります。

---

これらのオプションの詳細については、『*Framework 7.6 Deployment Guide*』の構成履歴ログについての項、および『*Framework 7.6 Configuration Options Reference Manual*』の履歴ログについての項を参照してください。

## 再同期のユーザ要求

要求による再同期は、顧客が手動で再同期手順を実行した場合に発生します (手順の詳細については、[135 ページ](#)の「構成データの再同期の方法」を参照)。再同期を開始するように指示されると、ICON が Configuration Server のすべての構成データを要求し、これを永続キャッシュに格納します。同時に、動的通知など、ICON と Configuration Server 間の他のアクティビティはすべて無効になります。ICON が永続キャッシュの構成データを IDB に転送すると、Configuration Server からのリアルタイム通知のモニタを再開します。

---

## 再同期の動作

構成オブジェクト用と構成オブジェクトの関係用の2種類のテーブルに、ICON が構成データを格納します。したがって、構成データの各レコードが構成オブジェクト タイプとデータベース ID (DBID) の2つの属性によって定義されます。

再同期中に、ICON が Configuration Server のすべての構成データを要求し、ローカルキャッシュに格納します。それぞれの属性 (オブジェクト タイプと DBID) について、IDB に存在するかどうかチェックします。

- 存在しない場合は、IDB に新規レコードを追加する。
- 存在する場合は、オブジェクトの格納されたプロパティが異なっていないかチェックする。異なる場合は、IDB レコードを更新する。

---

**注：** IDB に存在するが、Configuration Database には存在しないアクティブなオブジェクトには、削除のマークが付きます。

---

再同期中には、ICON が、Configuration Server からの新しい構成オブジェクト更新に関するリアルタイム通知の受信を停止します。Configuration Server が通知をキューに格納し、同期が完了したら配信します。

[表 17](#) は、新規、更新、および削除された構成オブジェクトについて、再同期の際に ICON が IDB に記録する情報の一部を示しています。

- フィールド列は、ICON が IDB にデータを送信する際に、書き込む GC\_\* IDB テーブルの列を示します。
- GSYS\_EXT\_INT1 フィールドの値は、タイムスタンプ フラグの信頼性を示しています (値の説明については、[132 ページ](#)の表 19 を参照)。

表 17:IDB に格納されるオブジェクトの変更内容

フィールド	新規オブジェクト	更新されたオブジェクト	削除されたオブジェクト
STATUS	1 (アクティブ)	1 (アクティブ)	2 (非アクティブ)
CREATED	同期時刻		
DELETED	null	null	同期時刻
LASTCHANGED	同期時刻	同期時刻	同期時刻
GSYS_EXT_INT1	1		2または3

上記のテーブルに値が入力されない場合、ICONは関連するフィールドにすでに存在する値に変更を加えません。

表18は、新規、更新、および削除されたオブジェクトの関係について、再同期の際にICONがIDBに記録する情報の一部を示しています。LASTCHANGEDの情報は関係には関与しないため、テーブルに含まれていません。

表 18:IDB に格納されるオブジェクトの関係の変更内容

STATUS	新規のオブジェクトの関係	更新されたオブジェクトの関係	削除されたオブジェクトの関係
CREATED	1 (アクティブ)	1 (アクティブ)	2 (非アクティブ)
DELETED	同期時刻		
LASTCHANGED	null	null	同期時刻
GSYS_EXT_INT1	1		2または3

## 信頼性フラグ

信頼性フラグによって、下流のレポートアプリケーション(Genesys Info Mart など)がIDBの2つの個別のインスタンスから最も信頼性の高いデータを抽出することが可能になります。すべてのIDB構成レコードで、GSYS\_EXT\_INT1フィールドにデータの信頼性を示す値が格納されます(表19を参照)。

表 19:信頼性フラグの値

値	値が示す内容
0	作成または削除タイムスタンプが信頼できる。タイムスタンプは Configuration Server のリアルタイム通知または構成履歴ログによるもの。
1	作成タイムスタンプが信頼できない。タイムスタンプが初期データロード時刻または再同期時刻に一致している。
2	削除タイムスタンプが信頼できない。タイムスタンプが初期データロード時刻または再同期時刻に一致している。
3	作成および削除タイムスタンプが信頼できない。両方のタイムスタンプが初期データロード時刻または再同期時刻に一致している。

## 再同期に必要な時間

同期処理の完了までに必要な時間は、以下のような要因によって異なります。

- Configuration Server のパフォーマンス
- ネットワークのパフォーマンス
- Configuration Database のサイズ
- IDB をホストするデータベースのサイズ

通常の条件では、同期に数分かかります。

## IDB の再同期が必要な状況

場合によっては、IDB の構成データが疑わしいことを ICON が認識し、特別なログメッセージ(09-25131)を生成します。しかし、ほとんどの場合はこのようなメッセージが生成されないため、ユーザが構成データの再同期を行うかどうか決定する必要があります。

**注：** この特別なログメッセージを生成した場合は、ICON が構成の変更のモニタを停止して待機状態に移行し、再同期開始のユーザ コマンドを受け取るまでそのままになります。構成データの変更が失われることを回避するために、アラーム条件を設定することを強くお勧めします(「[アラーム条件の設定](#)」を参照)。再同期が完了すると、ICON が直ちに構成変更のモニタを再開します。

以下に示す *例外的な状況* をガイドラインとして、再同期が必要か判断してください。

- Configuration Database の RDBMS がクラッシュし、バックアップを使用して回復する必要がある。137 ページの「バックアップを使用した Configuration Database の復元」を参照。
- 避けられない原因で一定時間 ICON が停止したが、Configuration Server は動作していて、構成オブジェクトまたは構成オブジェクトと他のオブジェクトの関係に変更が加えられた。
- 誤って一定時間 ICON の構成追跡が無効にされ、その間に構成関連データに変更が加えられた。
- ICON が誤った Configuration Database に対して構成追跡を実行した (ユーザによるエラー)。
- ICON が構成データの不整合を検出し、メッセージをログに記録した (IDB のアップグレード後、最初にログされる不整合のメッセージは正常で、無視して問題ない)。
- 下流のレポートアプリケーション (Genesys Info Mart など) が、IDB の構成データの喪失または不整合を検出する。Genesys Info Mart をご使用の場合は、この例外的な状況の詳細について、Genesys Info Mart のマニュアルを参照してください。

## アラーム条件の設定

IDB の構成データに不整合の可能性があると ICON が判断すると、以下の標準レベルのログ イベントを生成します。

```
09-25131: Configuration data inconsistency is detected;  
reason: [reason]. Waiting for customer command... (構成  
データの不整合を検出しました。理由: [理由]。ユーザの指示を待ち  
ます...)
```

ICON が構成変更のモニタが停止して、待機状態に移行し、ユーザが再同期の指示をするまでそのままになります。

データの喪失を防ぐために、Solution Control Interface で、このログ イベントに基づいてアラーム条件を設定することを強くお奨めします。Alarm Condition Wizard (アラーム条件ウィザード) を使用してアラーム条件を作成する方法については、*Framework 7.6 Solution Control Interface* のヘルプを参照してください。

ウィザードの [アラーム検出] ページが表示されたら、[図 12](#) のように、[検出] でログ イベントを指定します。

**Alarm Condition Wizard**

**Alarm Detection**  
Specify the log event that will trigger this alarm.

An alarm can be triggered by any documented log event. Please enter the identification number of the log event that will trigger this alarm.

Detect event:  -

Sometimes, an active Alarm Condition can be canceled by another log event reported by the same source. If you would like to use this mechanism, please enter the identification number of the log event that will cancel this alarm.

Cancel event:  -

For detailed information about log events, click More Information.

< Back   Next >   Finish   Cancel

図 12: SCI を使用した ICON のアラーム検出イベントの指定

対応する [キャンセル] (消去) のイベントはありませんが、再同期に必要なデータをすべて取得すると、ICON がログ イベント 09-25017 を生成します。このログ イベントをクリアランス イベントとして使用できます (136 ページの「再同期の推奨事項」を参照)。さらに、アラームの [クリアランス タイムアウト] を設定して、指定した時間 (時単位) がすぎたら、アラーム条件を消去できます (図 13 を参照)。

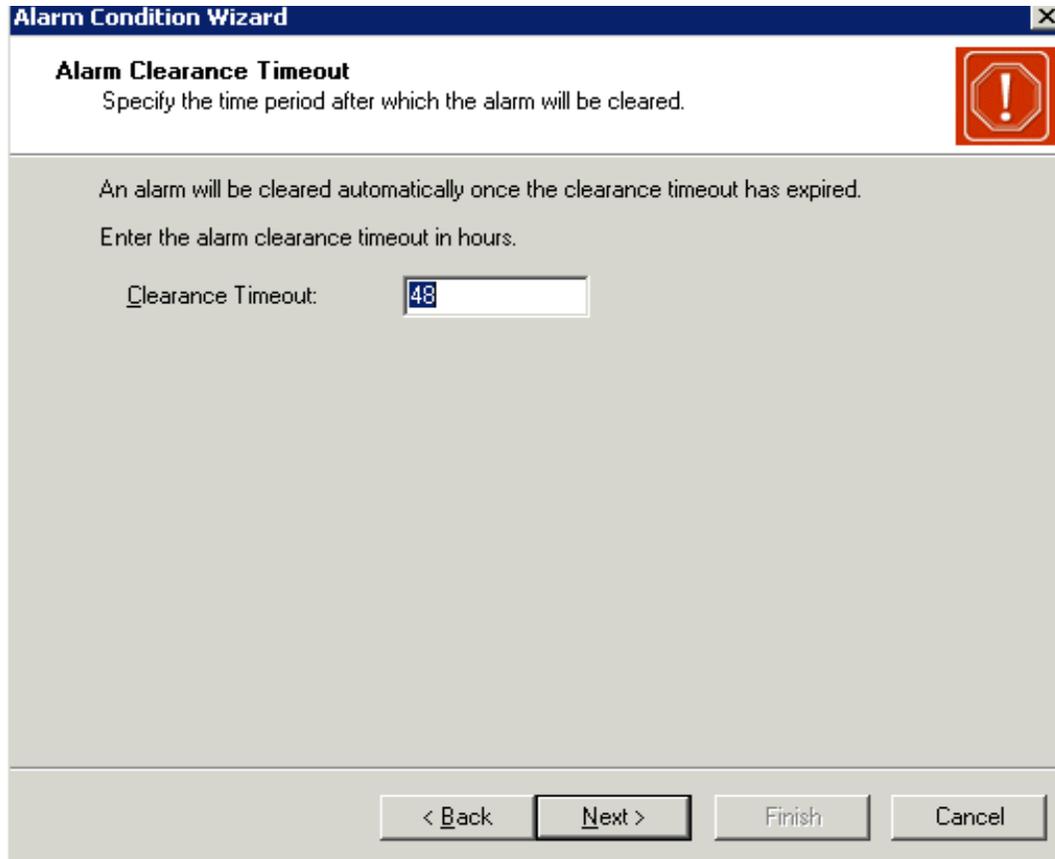


図 13: SCI を使用した ICON の [クリアランス タイムアウト] の指定

## 構成データの再同期の方法

IDB の構成データを再同期する必要がある場合は、ICON の実行中に以下の手順を行います。

1. role 構成オプションが cfg に設定された ICON アプリケーションが起動されていることを確認します。
2. Configuration Manager を使用して、この ICON アプリケーションの [アプリケーション プロパティ] ウィンドウで、start-cfg-sync オプションの値を 0 に設定します。
3. オプションプロパティのウィンドウで [OK] をクリックします。
4. ICON の [アプリケーション プロパティ] ウィンドウで [適用] をクリックして、設定を有効にします。
5. オプションの値を 1 に変更します。
6. [オプション プロパティ] ウィンドウで [OK] をクリックします。
7. ICON の [アプリケーション プロパティ] ウィンドウで [適用] をクリックして、新しい設定を有効にします。

これで、Interaction Concentratorが再同期プロセスを開始します。いったん開始すると、その後でオプション値を変更しても、完了するまで再同期プロセスが続行します。再同期が開始したときと終了したときに、ICONがメッセージをログに記録します。

**注：** start-cfg-sync オプションがあらかじめ 1 に設定されている場合や、ゼロ以外の値から 1 に設定した場合には、同期は開始しません。同期が開始するのは、このオプションが 0 から 1 に変更されたことを ICON が検知したときのみです。

## 再同期の推奨事項

- データの再同期中には、Configuration Server の動作モードを読み取り専用にすることを検討してください。

ICON が Configuration Server から再同期に必要なすべてのデータを取得する際に、以下の標準レベルのログ イベントを生成します。

09-25017 Configuration objects are reloaded in IDB  
(09-25017 構成オブジェクトが IDB に再ロードされます)

この時点で、Configuration Server の動作モードを通常に戻すことができます。再同期のアラーム条件を設定する際に、このログ イベントをクリアランス イベントとして使用することをお奨めします (133 ページの「アラーム条件の設定」を参照)。

- 下流のレポート アプリケーションが Interaction Concentrator をデータ ソースとして使用している場合は、再同期プロセス後に初めて ETL エンジンを実行する前に、以下の手順で、IDB のデータの準備が整っていることを確認します。
  - ICON ログでログ イベント「09-25017, Configuration objects are reloaded in IDB (09-25017, 構成オブジェクトが IDB に再ロードされます)」を確認する。
  - IDB に対して以下の SQL 文を実行する。

```
select eventid from G_SYNC_CONTROL where providertag = 5
```

上記の文がレコードを返さない場合、または eventID = 0 を返した場合は、まだ再同期が進行中です。

上記の文が eventID としてゼロ以外の値を返した場合は、再同期が完了しており、ETL エンジンを実行しても安全です。

バックアップの Configuration Database からデータを復元する場合 (たとえば、プライマリ Configuration Database が破損した場合は、以下の手順を行います。

- バックアップ コピーから Configuration Database を復元します (「バックアップを使用した Configuration Database の復元」を参照)。
- Interaction Concentrator を再起動します。
- IDB の構成データを手動で再同期します (135 ページの「構成データの再同期の方法」を参照)。

---

# バックアップを使用した Configuration Database の復元

バックアップから Configuration Database を復元するには、以下の手順を行います。

1. Configuration Server を停止します。
2. 構成データの変更を追跡している Interaction Concentrator インスタンス (role = cfg) を停止します。
3. バックアップから Configuration Database を復元します。
4. Configuration Server が、ICON に前に通知されていた同じ DBID を再使用しないように、Configuration Database で変更を行います (「[Configuration Database の変更](#)」を参照)。
5. Configuration Server を起動します。

## Configuration Database の変更

Configuration Server が同じ (すでに通知された) DBID を再使用することを避けるために、Configuration Database で以下の変更を行います。

---

**注：** ICON が格納する各タイプの構成オブジェクトについて、以下のステップを繰り返す必要があります (表 20 を参照)。

---

1. IDB に対して以下の SQL 文を実行して、Configuration Server が ICON に通知した最後のオブジェクト DBID を確認します (結果を保存する)。

```
select max(id) from <IDB_TABLE_NAME>
```

---

**注：** SQL 文の <IDB\_TABLE\_NAME> を適切なテーブル名と置き換えます (表 20 を参照)。

---

2. ステップ 1 で実行した文の結果が Null でない場合 (IDB テーブルにレコードがある場合) は、ステップ 3 に進みます。結果が Null の場合は、表 20 に示した次のタイプの構成オブジェクトに対して、ステップ 1 を繰り返します。
3. Configuration Database に対して以下の SQL 文を実行して、最後のオブジェクト DBID の最大値を確認します。

```
select MAX_DBID from CFG_MAX_DBID where object_type = <cfg_object_type>
```

---

**注：** SQL 文の <cfg\_object\_type> を Configuration Server のオブジェクトタイプの整数コードと置き換えます (表 20 を参照)。

---

4. **ステップ 3** で実行した文の結果が任意の値 (Null でない値) の場合は、**ステップ 6** に進みます。結果が Null の場合は、**ステップ 5** に進みます。

5. Configuration Database に対して以下の SQL 文を実行して、CFG\_MAX\_DBID という Configuration Database テーブルにレコードを挿入します。

```
insert into CFG_MAX_DBID (object_type,max_dbid) values
(<cfg_object_type>, <max_id> + 1)
```

**注:** SQL 文の <cfg\_object\_type> を Configuration Server のオブジェクトタイプの整数コードと置き換えます (表 20 を参照)。

SQL 文の <max\_id> を **ステップ 1** の max(id) の値と置き換えます。

6. Configuration Database に対して以下の SQL 文を実行して、CFG\_MAX\_DBID という Configuration Database テーブルのレコードを更新します。

```
update CFG_MAX_DBID set max_dbid = <max_id> + 1 where
object_type = <cfg_object_type>
```

**注:** SQL 文の <cfg\_object\_type> を Configuration Server のオブジェクトタイプの整数コードと置き換えます (表 20 を参照)。

SQL 文の <max\_id> を **ステップ 1** の max(id) の値と置き換えます。

7. ICON が格納する構成オブジェクトの各タイプに対して、**ステップ 1 ~ 6** を繰り返します。

表 20: Configuration Server オブジェクトのタイプと IDB テーブル

オブジェクトタイプ	IDB テーブル名: <IDB_TABLE_NAME>	Config Server オブジェクト タイプ: <cfg_object_type>
交換機	GC_SWITCH	1
エンドポイント(DN)	GC_ENDPOINT	2
構成員(エージェント)	GC_AGENT	3
作業場所	GC_PLACE	4
グループ	GC_GROUP	5
テナント	GC_TENANT	7
アプリケーション	GC_APPLICATION	9
スクリプト	GC_SCRIPT	12
スキル	GC_SKILL	13

表 20: Configuration Server オブジェクトのタイプと IDB テーブル (つづき)

オブジェクト タイプ	IDB テーブル名 : <IDB_TABLE_NAME>	Config Server オブジェクト タイプ : <cfg_object_type>
アクション コード	GC_ACTION_CODE	14
エージェント ログイン	GC_LOGIN	15
フォルダ	GC_FOLDER	22
テーブル フィールド	GC_FIELD	23
テーブル フォーマット	GC_FORMAT	24
テーブル アクセス	GC_TABLE_ACCESS	25
コーリング リスト	GC_CALLING_LIST	26
キャンペーン	GC_CAMPAIGN	27
トリートメント	GC_TREATMENT	28
フィルタ	GC_FILTER	29
タイム ゾーン	GC_TIME_ZONE	30
音声プロンプト	GC_VOICE_PROMPT	31
IVRポート	GC_IVRPORT	32
IVR	GC_IVR	33
ビジネス属性	GC_BUS_ATTRIBUTE	35
属性値	GC_ATTR_VALUE	36
目標表	GC_OBJ_TABLE	37





# 14 特別なストアド プロシージャの使用

Interaction Concentrator (ICON)が内部で使用して機能を実行するストアド プロシージャとは別に、Interaction Concentratorはユーザに関連するストアド プロシージャも備えています。

この章では、ICON ユーザに特に関係があるストアド プロシージャと、これを設定して実行する手順について説明します。サポートされる RDBMS それぞれについての例も取り上げます。

この章には以下の項があります。

- マージストアド プロシージャ (141 ページ)
- パージストアド プロシージャ (152 ページ)
- gsysInitTimeCode ストアド プロシージャ (161 ページ)
- カスタム ディスパッチャ (162 ページ)

---

**警告!** この章に記載された SQL に似た文は本当の SQL 文ではなく、取り上げるストアド プロシージャの論理を説明するためのものです。

---

---

## マージ ストアド プロシージャ

マージ ストアド プロシージャによって、Interaction Database (IDB)の音声インタラクション レコードを結合し、閉じたシングルサイト インタラクションとマルチサイト インタラクションのデータ処理をファイナライズします。このプロシージャは1つのIDBに格納された音声インタラクション レコードをマージします(同一IDB内マージ)。複数のIDBに分散したレコードはマージされません(IDB間マージまたは複数IDB間マージ)。

ICONプロセスのインスタンスがデータをIDBに書き込んでいるときに、バックグラウンドでマージプロシージャを実行できます。

**スケジューリング** 必要に応じて、顧客(通常はデータベース管理者)がマージ プロシージャを実行する必要があります。スケジュールを設定して、定期的にマージストアド プロシージャを実行することをお奨めします。スケジュール(15分未満の間隔)を設定しない場合は、IDBからデータを抽出する前に必ず実行してください。

---

**注：** Genesys Info Mart を使用する場合は、このプロシージャを 5 分ごとに実行してください。

---

マージ プロシージャを特別な場合にのみ実行すると、定期的に実行する場合と較べてかなり時間がかかります。さらに、この場合、プロシージャの実行中にICONのパフォーマンスが低下することがあります。

この項では、マージ プロシージャについて以下の項目を説明します。

- gsysIRMerge と gsysIRMerge2 スストアド プロシージャの機能
- マージ プロシージャの設定 (146 ページを参照)。
- マージ プロシージャのパラメータ (148 ページを参照)。
- マージ プロシージャの実行 (149 ページを参照)。

## gsysIRMerge と gsysIRMerge2

リリース 7.5 以降の Interaction Concentrator は、音声データ マージのために以下の 2 つのストアド プロシージャをサポートします。

- gsysIRMerge—gsysIRMerge2 のラップ
- gsysIRMerge2—実際にマージを実行するストアド プロシージャ

gsysIRMerge は、Interaction Concentrator 7.2 との下位互換性を維持するために用意されています。gsysIRMerge 7.2 の呼び出しは gsysIRMerge 7.5 以上の呼び出しと互換性があります。

### プロシージャに関連するテーブル

図 14 は、マージ プロシージャと関連するテーブルを示しています。

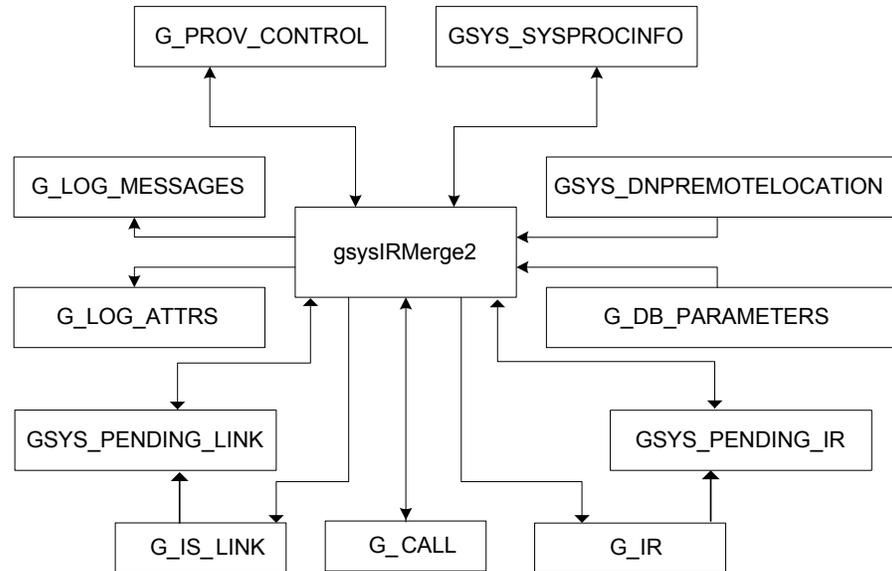


図 14: マージ プロシージャに関連するテーブル

以下のサブセクションで、各テーブルおよび各テーブルがマージ プロシージャで使用される方法について説明します。

### G\_LOG\_MESSAGES と G\_LOG\_ATTRS

#### G\_LOG\_MESSAGES

G\_LOG\_MESSAGES テーブルには、マージ プロシージャの開始(MESSAGE\_ID = 5020)と終了(MESSAGE\_ID = 5030)に関する情報が格納されます。

#### G\_LOG\_ATTRS

G\_LOG\_ATTRS テーブルには、エラーの説明(コードとメッセージ)および処理されたデータに関する詳細情報が格納されます。

Genesys Solution Control Interface (SCI)を使用して、この2つのテーブルの情報を読み取ることができます。SCIはGenesys Management LayerのGUI (Graphical User Interface : グラフィカル ユーザ インタフェース)コンポーネントです。

### G\_PROV\_CONTROL と GSYS\_SYSPROCINFO

#### G\_PROV\_CONTROL

マージ プロシージャがG\_PROV\_CONTROL テーブルから、データベースに情報を書き込んでいるICONインスタンスについての情報および最新のトランザクションに対応する数値を読み取ります。

プロシージャがG\_PROV\_CONTROL テーブルに対して以下のクエリを実行します。

```

SELECT PRIMARYID SYS_ID      /* ICON instance identity*/
   SEQCURRENT                /* number of latest
                             transaction*/
FROM G_PROV_CONTROL
   WHERE PROVIDERTAG = 1 AND SEQCURRENT IS NOT NULL
  
```

さらに、以下のクエリを使用して、このテーブルからマージ プロシージャ自身の最新トランザクションの数値を読み取ります。

```
SELECT SEQCOUNTER FROM G_PROV_CONTROL
WHERE DOMAINID = 0
AND PRIMARYID = 99
AND PROVIDERTAG = 0
```

この後で、マージ プロシージャがデータ処理を開始します。データ処理が完了すると、マージ プロシージャが自身のトランザクションの数値を新しい値に更新します。

### GSYS\_ SYSPROCINFO

マージ プロシージャがGSYS\_SYSPROCINFOテーブルから、処理済みの最新ICONトランザクションの数値についての情報を読み取ります。各ICONインスタンスに対して、以下のクエリが使用されます。

```
SELECT SEQPROCESSED FROM GSYS_SYSPROCINFO
WHERE PROVIDERTAG = 1 AND PROCNAME = 'gsysirmerge'
```

データ処理が完了すると、マージ プロシージャがトランザクションの数値を処理済みのものに更新します。

## GSYS\_DNPREMOTELOCATION と G\_DB\_PARAMETERS

GSYS\_DNPREMOTELOCATION と G\_DB\_PARAMETERS テーブルのデータを使用して、**滞留IS (Inter-Server) Link**を処理します。滞留IS Linkとは、別のサイトの対応するIS Linkに関する情報がないリンクです。

通常の状態では、IDBにインタラクショナルのすべての部分に関する完全な情報が格納されている場合に、マージ プロシージャがインタラクショナルをマージします。ただし、複数サイトに渡るインタラクショナルについては、同じIDBに書き込むICONインスタンスによってすべてのサイトがモニタされていない限り、IDBが十分な情報を入手して、インタラクショナルが終了したと判断し、インタラクショナルの構成要素を特定することはできません。このため、IS-Linkが滞留します。

### GSYS\_ DNPREMOTE LOCATION

マージ プロシージャがGSYS\_DNPREMOTELOCATIONテーブルから、IDBの特定のインスタンスに書き込むICONインスタンスによってモニタされていない交換機の名前を読み取ります。このデータベース側からは、モニタされていない交換機はリモート ロケーションであるとみなされます。

GSYS\_DNPREMOTELOCATIONテーブルのレコードは、マージ プロシージャに対して、リモート ロケーションを指示するIS-Linkが特定のIDBのインスタンス内でペアになっていないことを示します。したがって、マージ プロシージャは、IS-Linkの他方のインタラクショナルについて、それ以上情報を入手できないとみなします。このように、GSYS\_DNPREMOTELOCATIONテーブルのデータによって、マージ プロシージャがIS-Linkのタイムアウトがすぎるまで待つ必要がなくなるため、滞留IS-Linkを処理する時間が最小限になります (IS-Linkのタイムアウトの詳細については、「[G\\_DB\\_PARAMETERS](#)」を参照。146ページのstuckthresholdも参照)。

### G\_DB\_ PARAMETERS

マージ プロシージャが G\_DB\_PARAMETERS テーブルから、別のサイトのIS-Linkからの情報がレポートされるのを待つ時間間隔 (秒単位) を読み取ります。以下のクエリが使用されます。

```
SELECT VAL FROM G_DB_PARAMETERS
```

```
WHERE SECT = 'merge' AND
      OPT = 'stuckthreshold'
```

マージ プロシージャは G\_DB\_PARAMETERS テーブルの他のパラメータも使用します。詳細については、[146 ページ](#)の「IS-Link のタイムアウトとその他のパラメータ」を参照してください。

## GSYS\_PENDING\_IR と GSYS\_PENDING\_LINK

GSYS\_PENDING\_IR と GSYS\_PENDING\_LINK テーブルは、処理中のインタラクションレコードと IS-Link のマージ状態を追跡します。マージ プロシージャが GSYS\_PENDING\_IR と GSYS\_PENDING\_LINK テーブルに一時データを入力します。

## G\_IS\_LINK、G\_IR、G\_CALL

### G\_IS\_LINK

マージ プロシージャが G\_IS\_LINK テーブルを使用して、マルチサイトインタラクションを判定します。以下のいずれかが発生すると、選択されたマルチサイトインタラクションが完了したとみなされます。

- すべての IS-Link が別のサイトの対応する IS-Link を持つ。
- モニタされない交換機に対して IS-Link が開いている。
- 別のサイトから IS-Link が到着するまでのタイムアウトがすぎ、対応するすべての IR が閉じる。

マルチサイトインタラクションが完了すると、プロシージャが以下の更新を実行します。

```
UPDATE G_IS_LINK SET MERGESTATE = 3
      WHERE LINKID in (ALL IS LINK IN INTERACTION)
```

```
UPDATE G_IR SET MERGESTATE = 3
      ROOTIRID = (FIRST IR IN INTERACTION)
      GSYS_MSEQ = (the procedure's current TRANSACTION ID)
      GSYS_MSEQ_TS = (time of TRANSACTION)
      WHERE ROOTIRID in (ALL ROOT IRs IN INTERACTION)
```

```
UPDATE G_CALL
      SET ROOTIRID = (FIRST IR IN INTERACTION)
      WHERE ROOTIRID in (ALL ROOT IRs IN INTERACTION)
```

シングルサイトインタラクションの場合は、対応するすべての IR が閉じると、インタラクションが完了し、プロシージャが以下の更新を実行します。

```
UPDATE G_IR SET MERGESTATE = 3
      GSYS_MSEQ = (the procedure's current TRANSACTION ID)
      GSYS_MSEQ_TS = (time of TRANSACTION)
      WHERE IRID in (IRs IN INTERACTION)
```

## マージ プロシージャの設定

マージ プロシージャを設定する際には、IDB の以下の情報が目的に合うことを確認してください。

### モニタされない 交換機

- GSYS\_DNPREMOTELOCATION テーブルには、特定の IDB に書き込む ICON インスタンスによってモニタされない交換機の名前が格納されます。必要ならば、標準の INSERT 文を使用して、手動で GSYS\_DNPREMOTELOCATION テーブルを更新します。GSYS\_DNPREMOTELOCATION テーブルの REMOTELOCATION 列には、モニタされない交換機の名前が格納されます。

### 例

たとえば、2 つの交換機があって、それぞれが独自の IDB に書き込む個別の ICON によってモニタされているとします。

- ICON1 が交換機 SITE1\_sw1 をモニタし、IDB1 に書き込む。
- ICON2 が交換機 SITE2\_sw2 をモニタし、IDB2 に書き込む。

マージストアード プロシージャのパフォーマンスが最適になるように、それぞれの IDB に以下のレコードを追加します。

- IDB1 で、  
GSYS\_DNPREMOTELOCATION.REMOTELOCATION='SITE2\_sw2' を設定する。
- IDB2 で、  
GSYS\_DNPREMOTELOCATION.REMOTELOCATION='SITE1\_sw1' を設定する。

### IS-Link のタイム アウトとその他の パラメータ

- G\_DB\_PARAMETERS テーブルには、マージ プロシージャの動作を制御するために使用するパラメータが格納されます。表 21 に、パラメータとマージ プロシージャが使用するデフォルト値を示します。

表 21: G\_DB\_PARAMETERS テーブルのマージパラメータ

OPT 列	VAL 列			説明
	Oracle	Microsoft SQL	DB2	
nodnrl	0	0	0	マージ プロシージャがリモート ロケーションを無視するかどうかを指定。 有効値: 1 REMOTELOCATION を無視する。 0 REMOTELOCATION を無視しない。
stuckthreshold	28860	28860	28860	別のサイトから IS-Link 情報が送信されると予期される時間間隔(秒単位)。この時間がすぎると、IS-Link が滞留しているとみなされる。

注: すべてのエントリが列 SETID = 0 および列 SECT = 'merge' を持つ。

表 21:G\_DB\_PARAMETERS テーブルのマージパラメータ (つづき)

OPT列	VAL列			説明
	Oracle	Microsoft SQL	DB2	
step	75	75	25	マージ プロシージャが一度に選択するトランザクションの数(新規IRと新規リンクを読み取る際)。 小さい値を指定すると、コア テーブルのロックに遭遇することを制限可能。ただし、値が小さすぎると繰り返しが多くなる。 Microsoft SQLを使用する大規模な構築の詳細については、「 <a href="#">Microsoft SQLを使用する大規模構築の注意点</a> 」を参照。
limit	1000	2000	1000	マージ プロシージャが一度に閉じる対象とみなすルートインタラクションの数 (G_IS_LINK、G_IR、G_CALLテーブルを更新する際)。 MSEQ当たりのIRの数を効果的に制限可能。ただし、値が小さすぎると繰り返しが多くなる。 Microsoft SQLを使用する大規模な構築の詳細については、「 <a href="#">Microsoft SQLを使用する大規模構築の注意点</a> 」を参照。
limit2	10000	10000	10000	マージ プロシージャが閉じる段階を開始する前に読み取る新規リンクとインタラクションレコードの数 (G_IS_LINK、G_IR、G_CALLテーブルを更新する際)。 非常に大きい数を指定すると、パフォーマンス低下の可能性はある。
注: すべてのエントリが列SETID = 0および列SECT = 'merge'を持つ。				

### Microsoft SQL を使用する大規模構築の注意点

Microsoft SQL の場合、step と limit パラメータのデフォルト値では、大量のデータ (数百万のインタラクション) を格納する IDB には不十分です。最適なパフォーマンスを得るために、まず、以下の値を使用してみてください。

- step パラメータの値を 200 に増加する。
- limit パラメータの値を 3000 に増加する。

ただし、ご使用の大規模構築に最適な値を見つけるために、いろいろな値を試すことをお勧めします。

## G\_DB\_PARAMETERS テーブルの更新

必要に応じてG\_DB\_PARAMETERSテーブルを更新してください。このために、Interaction ConcentratorにはsvcUpdateDBParametersというストアド プロシージャが用意されています。このストアド プロシージャにSETID値(常に0)、SECT値(常に 'merge')、OPT値(146ページの表21のOPT列を参照)、および VAL値を設定する必要があります。

たとえば、IS-Linkのタイムアウトをデータベースに書き込むには、以下の文を実行します。正確な構文はRDBMSによって異なります。

```
EXEC svcUpdateDBParameters
0,
'merge',
'stuckthreshold',
'<TIMEOUT>'
```

## gsysIRMerge2 パラメータ

gsysIRMerge2ストアド プロシージャでは、2つの入力パラメータと3つの出力パラメータの合計5つを使用できます。

gsysIRMergeストアド プロシージャに対する呼び出しにより gsysIRMerge2ストアド プロシージャが呼び出されると、gsysIRMergeが必要なパラメータを提供します。

表22は、gsysIRMerge2の入力パラメータと出力パラメータ、および gsysIRMergeから gsysIRMerge2が呼び出されたときに入力パラメータに使用される値を示しています。

表 22: マージ プロシージャのパラメータ

パラメータ	データ型	説明
<b>入力</b>		
OVERRIDE	int	ロックのオーバーライド設定。 有効値: 0 ロックをオーバーライドしない。 1 PREVCALLER==CALLERの場合にオーバーライド。 2 常にオーバーライド。 gsysIRMergeの値: 0
CALLER	varchar	ストアド プロシージャの呼び出し側の名前。 有効値: 任意の文字列 (Infomart_DBID_1001, DCA6など)。 gsysIRMergeの値: 'singleIDBMerge'

表 22: マージ プロシージャのパラメータ (つづき)

パラメータ	データ型	説明
<b>出力</b>		
RESULT	int	ストアドプロシージャ呼び出しの結果。 有効値: 0 成功。 2 IDBがロックされている。 1 その他のエラー。 gsysIRMergeの値: RESULT
PREVCALLER	varchar	RESULT==2の場合に前の呼び出し側の名前を指定する。 gsysIRMergeの値: PREVCALLER
PREVAGE	int	RESULT==2の場合に、前の呼び出し側がロックを取得した後の時間(秒単位)を指定する。 gsysIRMergeの値: PREVAGE

RESULTパラメータは、プロシージャの実行状況に関する情報を提供します。他のパラメータは同時処理の制御に使用され、所定の時間にプロシージャの複数のインスタンスが実行されることを防ぎます。

**注:** gsysIRMerge スストアドプロシージャは出力パラメータ (RESULT、PREVCALLER、PREVAGE) を破棄します。

## マージ プロシージャの実行

マージストアドプロシージャを定期的に行うことをお奨めします。これによって、データ処理に必要な時間と他のアプリケーション(下流のレポートシステムなど)に正しいデータを配信するために必要な時間を短縮できます。マージストアドプロシージャは以下の2つの方法で呼び出すことができます。

- gsysIRMerge を呼び出す。
- gsysIRMerge2 を呼び出す。

以下のサブセクションでそれぞれの方法について説明します。

**注:** マージ プロシージャは音声インタラクション データに対してのみ実行できます。

## gsysIRMerge の呼び出し

マージ プロシージャを実行するには、以下の文を使用します。正確な構文は RDBMS によって異なります。

```
EXEC gsysIRMerge
```

次に、gsysIRMergeストアード プロシージャが、パラメータ1、`singleIDBMerge`、RESULT、PREVCALLERおよびPREVAGEを使用して、gsysIRMerge2を呼び出します。

## gsysIRMerge2 の呼び出し

gsysIRMerge2ストアード プロシージャを直接呼び出して、入力パラメータと出力パラメータに独自の値を指定できます。プロシージャを呼び出すときに、必要なパラメータを指定する必要があります(入力および出力パラメータの詳細については、[148ページ](#)の「gsysIRMerge2パラメータ」を参照)。

gsysIRMerge2 ストアード プロシージャを直接呼び出すスクリプトには、サポートされる RDBMS に応じてそれぞれ異なる構文上の要件があります。この項では以下の例を取り上げます。

- Oracle の例
- Microsoft SQL の例
- DB2 の例

### Oracle の例

```
declare
  OVERRIDE$ int := 1;
  CALLER$ varchar2(40) := 'singleIDBMerge';
  RESULT$ int;
  PREVCALLER$ varchar2(40);
  PREVAGE$ int;
begin
  gsysIRMerge2( OVERRIDE$, CALLER$, RESULT$,
    PREVCALLER$, PREVAGE$ );
  dbms_output.put_line('='||RESULT$);
end;
```

### Microsoft SQL の例

```
declare @OVERRIDE          int
declare @CALLER            varchar(40)
declare @RESULT            int
declare @PREVCALLER        varchar(40)
declare @PREVAGE           int
set @OVERRIDE = 1
set @CALLER = 'singleIDBMerge'
exec gsysIRMerge2 @OVERRIDE, @CALLER, @RESULT,
@PREVCALLER, @PREVAGE
```

## DB2 の例

```
create procedure gsysIRMergeTest
DYNAMIC RESULT SETS 1
language SQL
begin
    declare OVERRIDE          int default 1;
    declare CALLER            varchar(40) default
'singleIDBMerge';
    declare RESULT            int;
    declare PREVCALLER        varchar(40);
    declare PREVAGE           int;

    call gsysIRMerge2( OVERRIDE, CALLER, RESULT,
PREVCALLER, PREVAGE );
    begin
        declare c_cur cursor with return for
            select RESULT, PREVCALLER, PREVAGE from
sysibm.sysdummy1;
        open c_cur;
    end;

end;
call gsysIRMergeTest;
drop procedure gsysIRMergeTest;
```

## マージ プロシージャのパフォーマンス向上

データベース構成、データベース設定、およびマージ プロシージャのスケジューリングが、マージ プロシージャのパフォーマンスに大きく影響することがあります。

デッドロックの発生を最小限に抑え、マージ プロシージャのパフォーマンスを最適化するために、[146ページ](#)の「マージ プロシージャの設定」および[171ページ](#)の「マージ プロシージャの実行に時間がかかるまたは安定しない」の内容を十分に確認してください。

### 統計の計算

定期的に統計を収集することは、データベースのパフォーマンスを良好にするために一般的な要件です。マージ プロシージャの最適なパフォーマンスのために、使用可能なデータベース統計が適切な表現である(比較的新しい)必要があります。

## マージ プロシージャのトラブルシューティング

マージ プロシージャの実行に関する問題とその解決方法については、[168 ページ](#)の「マージ プロシージャの問題」を参照してください。

---

**注：** デッドロックの発生を最小限に抑え、マージ プロシージャのパフォーマンスを最適化するために、[146 ページ](#)の「マージ プロシージャの設定」および [171 ページ](#)の「マージ プロシージャの実行に時間がかかるまたは安定しない」の内容を十分に確認してください。ただし、デッドロックは避けられないため、下流のレポート アプリケーションがデッドロックを処理できる必要があります。

---

プロシージャに例外が発生した場合は、接続に対するアクションを実行する前に、rollback 文を発行してください。

---

## ページ ストアド プロシージャ

Interaction Concentrator のパフォーマンスに影響する最も大きな要因が IDB のサイズです。IDB を管理しやすい状態に維持するために、適切なページ ストラテジが必要です。

Interaction Concentrator には、IDB の音声データをページするためのストアド プロシージャが用意されています。ページ プロシージャには以下のような機能があります。

- マージした音声インタラクション データを IDB から安全にページする。同時に、論理的に関連するすべての情報が削除されます。
- 動作中に、ICON のパフォーマンスに影響を与えない。
- 削除するデータの範囲を制限する入力パラメータを使用可能。

---

**注：** Interaction Concentrator 7.5 および 7.6 では、テナント ID によるデータのページはサポートされません。

---

この項では、ページ プロシージャについて以下の項目を説明します。

- ページ プロシージャの詳細 ([「ページ ストアド プロシージャ」](#) を参照)
- ログとエラー処理 ([156 ページ](#) を参照)
- ページ プロシージャのスケジューリング ([157 ページ](#) を参照)
- ページ プロシージャの設定 ([158 ページ](#) を参照)
- ページ プロシージャの実行 ([160 ページ](#) を参照)

---

**注：** ページ ストアド プロシージャは、マージされた音声インタラクション 関連のデータのみをページします。マルチメディア インタラクションの電子メール チェーンと保持の要件は、個々の構築のビジネス モデルによって異なります。したがって、マルチメディア インタラクションを安全にページする汎用プロシージャを提供することはできません。

---

## ページストアド プロシージャ

IDB からデータを削除するためのプライマリ ストアド プロシージャを以下に示します。

- gsysPurgeIR—IR (Interaction Record : インタラクション レコード ) をパージする。
- gsysPurgeUDH—UDH (User Data History : ユーザ データ履歴 ) データをパージする (154 ページの「ユーザ データ履歴のパージ」を参照)。
- gsysPurgeLS—ALS (Agent Login Session : エージェント ログイン セッション) データをパージする (155 ページの「エージェント ログイン セッションのパージ」を参照)。
- gsysPurgeOS—アウトバウンド データをパージする (155 ページの「アウトバウンドセッションのパージ」を参照)。

プライマリ ストアド プロシージャはすべて同時使用に対して保護されています。一度に実行できるのは、特定のプライマリ プロシージャの1つのインスタンスだけです。

ICONプロセスのインスタンスがデータをIDBに書き込んでいるときに、バックグラウンドでパージプロシージャを実行できます。

**保持期間** すべてのプライマリ パージ プロシージャで、`purge stored procedures <count days>`という入力パラメータを使用して**保持期間**を指定できます(現在の日付より前の日数、この期間はデータがパージされない)。保持期間より前の日付のデータはパージ対象となります。入力パラメータは正の整数(最小値 = 1、最大値なし)の必要があります。

### IR のパージ

gsysPurgeIR は、IR をパージするためのプライマリ プロシージャで、以下のストアド プロシージャを呼び出します。

- gsysPurge\_GCC—IR データを削除する。
- gsysPurge\_GUD—ユーザ データを削除する。

gsysPurgeIRはマージされたIRのみを削除します。

**入力パラメータ** gsysPurgeIRの入力パラメータ<count days>は、IR関連データの保持期間を計算するために使用されます。マージされたIRのうち、現在の日付マイナス<count days>より前のマージ日付を持つIRのすべてのIR関連データが、パージプロシージャによって削除されます。現在の日付は深夜12時に開始し、現在の日付の時間は無視されます。たとえば、ストアド プロシージャが呼び出された日付と時刻が05/05/2007 13:15で、入力パラメータが1の場合は、05/04/2007 00:00より前のマージ日付を持つすべてのIRが削除されます。

## IR パージ プロシージャに関連するテーブル

以下のテーブルには IR 関連データが格納されます。構築によっては、一部のテーブルに入力されない場合もあります。

- G\_IR
- G\_IR\_HISTORY
- G\_CALL
- G\_CALL\_HISTORY
- G\_PARTY
- G\_PARTY\_HISTORY
- G\_IS\_LINK
- G\_IS\_LINK\_HISTORY
- G\_ROUTE\_RESULT
- G\_VIRTUAL\_QUEUE
- G\_CALL\_STAT
- G\_PARTY\_STAT
- G\_CALL\_USERDATA
- G\_CALL\_USERDATA\_CUST
- G\_CALL\_USERDATA\_CUST1
- G\_CALL\_USERDATA\_CUST2

IR パージ プロシージャが完了すると、テーブルに存在しないデータへの参照がなくなります。たとえば、G\_PARTY\_STAT テーブルには、G\_PARTY テーブルからパージされたパーティの参照は含まれません。

## ユーザ データ履歴のパージ

gsysPurgeUDH は、UDH (User Data History : ユーザ データ履歴) レコードをパージするためのプライマリ プロシージャです。このプロシージャはマージされた IR に関連するレコードのみを削除します。

### 入力パラメータ

gsysPurgeUDH の入力パラメータ <count days> は、UDH データの保持期間を計算するために使用されます。以下のいずれかの条件を満たすすべての UDH レコードが、パージ プロシージャによって削除されます。

- 対応する IR のマージ日付が、現在の日付マイナス <count days> より前の場合
- 対応する IR が存在しない、かつ履歴レコードが追加された日付が現在の日付マイナス <count days> より前の場合

現在の日付は深夜 12 時に開始し、現在の日付の時間は無視されます。

## UDH パージ プロシージャに関連するテーブル

以下のテーブルには UDH データが格納されます。IDB に書き込む ICON アプリケーションに対して構成した役割および添付データ仕様によっては、テーブルに入力されない場合もあります。

- G\_SECURE\_USERDATA\_HISTORY
- G\_USERDATA\_HISTORY

UDH パージ プロシージャが完了すると、テーブルに存在しないデータへの参照がなくなります。

## エージェント ログイン セッションのページ

gsysPurgeLS は、ALS (Agent Login Session : エージェント ログイン セッション) データをページするためのプライマリ プロシージャで、以下のストアド プロシージャを呼び出します。

- gsysPurge\_GLS— セッションを削除する。

gsysPurgeLSは閉じたセッションのみを削除します。

### 入力パラメータ

gsysPurgeLSの入力パラメータ<count days>は、ALS関連データの保持期間を計算するために使用されます。閉じたエージェント ログイン セッションのうち、現在の日付マイナス<count days>より前の終了日付を持つものすべてのセッション関連データが、ページ プロシージャによって削除されます。現在の日付は深夜12時に開始し、現在の日付の時間は無視されます。

### ALS ページ プロシージャに関連するテーブル

以下のテーブルには ALS 関連データが格納されます。

- G\_LOGIN\_SESSION
- G\_AGENT\_STATE\_HISTORY
- G\_AGENT\_STATE\_RC
- G\_DND\_HISTORY
- GS\_AGENT\_STAT
- GS\_AGENT\_STAT\_WM
- GX\_SESSION\_ENDPOINT

ALS ページ プロシージャが完了すると、テーブルの存在しないデータの参照がなくなります。ただし、存在しないIRの参照は存在する可能性があります。

## アウトバウンド セッションのページ

gsysPurgeOS は、OS (Outbound-Session : アウトバウンドセッション) データをページするためのプライマリ プロシージャで、以下のストアド プロシージャを呼び出します。

- gsysPurge\_GOS— アウトバウンドセッション関連のデータを削除する。

### 入力パラメータ

gsysPurgeOSの入力パラメータ<count days>は、アウトバウンドセッションデータの保持期間を計算するために使用されます。閉じたアウトバウンド キャンペーンセッションのうち、現在の日付マイナス<count days>より前の終了日付を持つものすべてのセッション関連データが、ページ プロシージャによって削除されます。現在の日付は深夜12時に開始し、現在の日付の時間は無視されます。

gsysPurgeOSは、閉じたキャンペーンセッションのみのアウトバウンドセッション関連データを削除します。親要素キャンペーンセッションが閉じていない場合は、対応するすべてのチェーンが削除されません。

## OS ページ プロシージャに関連するテーブル

以下のテーブルには OS 関連データが格納されます。

- GO\_CAMPAIGN
- GO\_CAMPPROP\_HIST
- GO\_METRICS
- GO\_CHAIN
- GO\_CAMPAIGNHISTORY
- GOX\_CHAIN\_CALL
- GO\_RECORD
- GO\_CHAINREC\_HIST
- GO\_CUSTOM\_FIELDS
- GO\_SECURE\_FIELDS
- GO\_SEC\_FIELDHIST
- GO\_FIELDHIST

OS ページ プロシージャが完了すると、テーブルに存在しないデータの参照がなくなります。ただし、存在しない IR の参照は存在する可能性があります。

## ログとエラー処理

### G\_LOG\_MESSAGES

G\_LOG\_MESSAGES テーブルには、ページ プロシージャの動作状態について、以下の情報のログが記録されます。

- purge stored procedures Started ( ページストアド プロシージャが開始 )—プライマリ プロシージャが開始した。
- ERROR ( エラー )—データベース エラーが発生した (G\_LOG\_ATTRS テーブルにエラーの詳細が記録される )。
- Completed [OK|ERROR|Locked] (完了 [OK|エラー|ロック])—プライマリ プロシージャが終了または停止した。Completed: Locked は、プロシージャの前のインスタンスがまだ実行中のため、プライマリ プロシージャが停止したことを意味する。

### G\_LOG\_ATTRS

G\_LOG\_ATTRS テーブルには、ページ プロシージャの動作について以下の情報が格納されます。

- エラーの詳細 ( コードとメッセージ )
- <count days> 入力パラメータの値
- メイン テーブルからページされたレコードの数 ( 各プライマリ ページ プロシージャ別 )、すべてのテーブルからページされたレコードの数など、処理されたデータに関する情報

Genesys Solution Control Interface (SCI) を使用して、G\_LOG\_MESSAGES および G\_LOG\_ATTRS テーブルの情報を読み取ることができます。SCI は Genesys Management Layer の GUI (Graphical User Interface : グラフィカル ユーザ インタフェース) コンポーネントです。

## 自動回復

ページ プロシージャはいくつかの一時テーブルを使用し、実行中にさらにインデックスを更新します。ページ プロシージャの実行中にエラーが発生すると、プロシージャが終了しますが、プロシージャをリセットするための特別なアクションは必要ありません。次にプロシージャを起動するときに、該当する

IDB テーブルの最小マージ シーケンスまたはタイムスタンプ値を使用してエントリ ポイントが計算されます。

**タイムアウト時間** 最大トランザクション時間のパラメータによって、プロシージャ ロックのタイムアウト時間が設定されます。処理されないエラーが原因でページ プロシージャが終了した場合に、タイムアウト時間がすぎた後でプロシージャの次のインスタンスが開始すると、ロックが無効になります。最大トランザクション時間のパラメータの詳細については、[158 ページ](#)の「ページ プロシージャの設定」を参照してください。

## ページ プロシージャのスケジューリング

必要に応じて、顧客(通常はデータベース管理者)がページ プロシージャを実行する必要があります。コンタクト センターのアクティビティが少ないときに、ページ プロシージャを実行することをお奨めします。

ページ プロシージャを特別な場合にのみ実行すると、定期的に行う場合と比べてかなり時間がかかります。さらに、この場合、プロシージャの実行中に ICON のパフォーマンスが低下することがあります。

ページ プロシージャを実行する順序に特に制限はありません。ただし、実行中に最適なパフォーマンスを確保し、データの整合性を保持するために、以下に示す順序で連続してページ プロシージャを実行することをお奨めします。

- gsysPurgeIR (IR のページ)
- gsysPurgeUDH (UDH データのページ)
- gsysPurgeLS (ALS データのページ)
- gsysPurgeOS (OS データのページ)

特に、1 日分を超える量のデータをページする場合は、ページ プロシージャを連続して実行してください。

さらに、以下のような考慮点と推奨事項にも注意してください。

- ページ プロシージャを連続して実行する際には、場合によっては、後半に実行されるページ プロシージャの <count days> 入力パラメータを調整する必要があります。深夜 12 時より後にプロシージャが開始することがあるためです。

たとえば、IR をページするプロシージャが 05/05/2007 20:00 (08:00 PM) に開始し、<count days> 入力パラメータが 1 の場合は、05/05/2007 23:59:59 (11:59:59 PM) より後に開始するすべてのプロシージャの入力パラメータを 2 に設定します。

- ページ プロシージャによってページされるのはマージされたインタラクションだけのため、ページ プロシージャをスケジューリングするには、マージ プロシージャのスケジュールを考慮する必要があります。たとえシングル サイト構築であっても、先にマージ プロシージャを実行しなければ、IDB からレコードをページできません。

- 下流のレポート アプリケーションの ETL (Extraction, Transformation, and Loading : 抽出、変換、およびロード) サイクルを考慮し、下流のレポート アプリケーションが正常にデータを抽出する前に、データが削除されることを防止する必要があります。

一般には、下流のレポート アプリケーションが定期的な ETL サイクルを完了した後で、ページ プロシージャを実行するようにスケジュールを設定します。十分な大きさのスライド ウィンドウにデータを保持し、ページ プロシージャを実行するたびに、最も古い ETL サイクルに相当するデータだけを削除してください。

下流のレポート アプリケーションが、複数の ETL サイクルのデータのバックログの抽出する機能を備えている場合があります (Genesys Info Mart の limit-extract-data 構成オプションなど)。この機能を使用する場合は、特定の ETL サイクルの終了時に、下流のレポート アプリケーションが、抽出時に入手可能だったすべてのデータを抽出できないことがあるため、注意してください。

---

**注：** Genesys Info Mart を下流のレポート アプリケーションとして使用している場合は、Genesys Info Mart の Job\_AggregateGIM および Job\_MaintainGIM ジョブと並行してページ プロシージャを実行できます。

---

- 下流のレポート アプリケーションが ETL ジョブ サイクルを正しく完了できない場合は、問題が解決し、ETL ジョブ サイクルが正常に再開するまで、ページ プロシージャを呼び出すことを中止する必要があります。そうでない場合、下流のレポート アプリケーションが抽出する前に、IDB ソース データが削除されることがあります。

## ページ プロシージャの設定

G\_DB\_PARAMETERS テーブルには、ページ プロシージャの動作を制御するためのパラメータが格納されます。ページ プロシージャを設定する際には、IDB のパラメータ設定がご使用の環境に適切か確認してください。

146 ページの表 21 に、パラメータとデフォルト値を示します。

表 23:G\_DB\_PARAMETERS テーブルのページパラメータ

適用される プロシージャ	G_DB_PARAMETERS テーブル		説明
	OPT 列	VAL 列	
gsysPurgeIR	IR_seq_step	75	データ検索に使用されるチャンク サイズ (ICON トランザクションの数)。
gsysPurgeUDH	UDH_time_step	300	1つのデータベース トランザクション内で ページする行数を計算するためのチャンク サ イズ(秒単位)。
gsysPurgeLS	LS_time_step	500	1つのデータベース トランザクション内で ページする行数を計算するためのチャンク サ イズ(秒単位)。
gsysPurgeOS	OS_time_step	500	1つのデータベース トランザクション内で ページする行数を計算するためのチャンク サ イズ(秒単位)。
gsysPurgeIR	IR_max_tran_time	600	プロシージャ ロックのタイムアウト時間(秒 単位)。最大トランザクション時間は、最も新 しいページ トランザクションから現在までの 時間。この時間がすぎると、プロシージャの 新規インスタンスの実行が許可される。
gsysPurgeUDH	UDH_max_tran_time		
gsysPurgeLS	LS_max_tran_time		
gsysPurgeOS	OS_max_tran_time		
注: すべてのエントリが列SETID = 0および列SECT = 'purge'を持つ。			

### G\_DB\_ PARAMETERS テーブルの更新

必要に応じて G\_DB\_PARAMETERS テーブルを更新してください。このために、Interaction Concentrator には svcUpdateDBParameters というストアド プロシージャが用意されています。このストアド プロシージャに SETID 値 (常に 0)、SECT 値 (常に 'purge')、OPT 値 (146 ページの表 21 の OPT 列を参照)、および VAL 値を設定する必要があります。

たとえば、IR パージプロシージャのプロシージャ ロック タイムアウトの値を変更するには、以下の文を実行します。正確な構文は RDBMS によって異なります。

```
EXEC svcUpdateDBParameters
0,
'purge',
'IR_max_tran_time',
'<TIMEOUT>'
```

## パージ プロシージャの実行

パージストアド プロシージャは毎日実行することをお奨めします。これによって、データ処理に必要な時間と他のアプリケーション(下流のレポートシステムなど)に正しいデータを配信するために必要な時間を短縮できます。スケジューリングのその他の考慮点については、[157ページ](#)の「パージ プロシージャのスケジューリング」を参照してください。

### パージストアド プロシージャの呼び出し

パージ プロシージャを呼び出す際には、<count days> 入力パラメータを指定する必要があります。

パージ プロシージャを呼び出すスクリプトの構文の要件は、サポートされる RDBMS によって異なります。この項では以下の例を取り上げます。

- Oracle の例
- Microsoft SQL の例 ([150 ページ](#)を参照)
- DB2 の例 ([151 ページ](#)を参照)

これらの例では、IRパージおよびALSパージ プロシージャを同じセッション内で連続して実行し、保持期間には異なる値を使用します。

#### Oracle の例

```
declare
  COUNT_DAYS$ int := 14;
begin
  gsysPurgeIR( COUNT_DAYS$ );
end;
```

```
declare
  COUNT_DAYS1$ int := 15;
begin
  gsysPurgeLS( COUNT_DAYS1$ );
end;
```

#### Microsoft SQL の例

```
declare @COUNT_DAYS          int
set @COUNT_DAYS = 14
exec gsysPurgeIR @COUNT_DAYS

declare @COUNT_DAYS1        int
set @COUNT_DAYS1 = 15
exec gsysPurgeLS @COUNT_DAYS1
```

**DB2 の例**

```

language SQL
begin
    declare COUNT_DAYS      int default 14;

    call gsysPurgeIR( COUNT_DAYS );

end;

language SQL
begin
    declare COUNT_DAYS1    int default 15;

    call gsysPurgeLS( COUNT_DAYS1 );

end;

```

---

## gsysInitTimeCode スストアド プロシージャ

ICON では `gsysInitTimeCode` スストアド プロシージャを使用して、`G_TIMECODE` テーブルを設定します。図 15 は、時間設定プロシージャに関連する 3 つのテーブルを示しています。

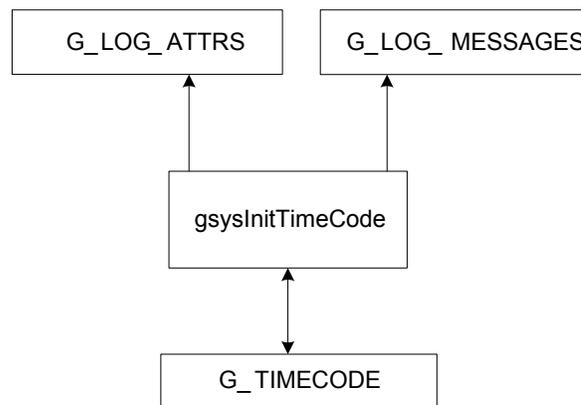


図 15: 時間設定プロシージャに関連するテーブル

以下のサブセクションで、各テーブルおよび各テーブルが時間設定プロシージャで使用される方法について説明します。

**G\_LOG\_MESSAGES と G\_LOG\_ATTRS****G\_LOG\_MESSAGES**

`G_LOG_MESSAGES` テーブルには、時間設定プロシージャの開始(`MESSAGE_ID = 5040`)および終了(`MESSAGE_ID = 5050`)についての情報が格納されます。また、プロシージャの実行中に発生したエラー (`MESSAGE_ID = 5045`) についての情報も格納されます。

**G\_LOG\_ATTRS** G\_LOG\_ATTRS テーブルには、エラーの説明(コードとメッセージ)および処理されたデータに関する詳細情報が格納されます。

Genesys SCI を使用して、この 2 つのテーブルの情報を読み取ることができます。

## G\_TIMECODE

時間設定プロシージャを実行すると、プロシージャの入力パラメータに基づいて G\_TIMECODE テーブルに入力されます。G\_TIMECODE テーブルを使用して、時間間隔レポートを作成できます。このテーブルのフィールド ID は他のテーブルの \*\_TCODE フィールドと関連し、1970 年 1 月 1 日以降、5 分間隔でカウントされた時間(秒単位)を表します。

## 時間設定プロシージャの設定

時間設定プロシージャを実行するための設定は必要ありません。

## 時間設定プロシージャの実行

以下の入力パラメータを使用して、必要な頻度で時間設定プロシージャを実行してください。

- BEGIN\_DATE— 最初の時間間隔の日付
- END\_DATE— 最後の時間間隔の日付

時間設定プロシージャを実行するには、以下の文を使用します。正確な構文は RDBMS によって異なります。

```
EXEC gsysInitTimeCode  
getdate(),  
getdate()+365
```

---

## カスタム ディスパッチャ

カスタマイズされた添付データの処理をサポートするために、IDB 初期化スクリプトによって、以下の 2 つのカスタム ディスパッチャストアードプロシージャが作成されます。

- gudCustDisp1
- gudCustDisp2

必要な添付データを格納するために、スクリプトを変更してカスタム ディスパッチャを作成する必要があります。

カスタム ディスパッチャの詳細については、[48 ページ](#)の「カスタム ディスパッチャ」を参照してください。



# 15

## ICON システムと構築に関する トラブルシューティング

この章では、Interaction Concentrator (ICON)アプリケーションの起動や実行の際に発生する問題と、その解決方法について説明します。以下の項があります。

- [起動の問題 \(163 ページ\)](#)
- [ランタイムの問題 \(165 ページ\)](#)
- [マージプロシージャの問題 \(168 ページ\)](#)

---

### 起動の問題

以下に起動に関するよくある問題を示します。

- ICON が Configuration Server に接続できない ([「Configuration Server に接続できない」](#) を参照)。
- 起動の際に ICON が終了する ([164 ページ](#)の「起動の際に ICON が終了する」を参照)。

### Configuration Server に接続できない

以下のような原因が考えられます。

- ICON アプリケーション オブジェクトの [サーバ情報] タブのコマンドラインパラメータで指定された Configuration Server のホストとポートが誤っている。

**解決方法:** コマンドラインパラメータを修正して、アプリケーションを再起動します。コマンドラインパラメータの詳細については、[98 ページ](#)の「コマンドラインパラメータ」を参照してください。

- Configuration Server が実行されていない、またはネットワークを介してアクセスできない。

**解決方法 :** Configuration Server を起動するか、ネットワーク接続を再度確立します。

## 起動の際に ICON が終了する

起動の問題について、ICON のログ ファイルを確認してください。以下のような原因が考えられます。

- ICON の起動コマンド ラインで指定されたアプリケーション名が、Configuration Layer の既存のアプリケーション オブジェクトと一致していない。

**解決方法 :** アプリケーション オブジェクトを作成します。ICON アプリケーション オブジェクトの作成と構成の詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のインストールと構成についての章を参照してください。

- ICON の起動コマンド ラインで指定されたアプリケーション名が、Call Concentrator アプリケーションタイプではないアプリケーション オブジェクトを参照している。

**解決方法 :** 誤ったタイプのアプリケーション オブジェクトを削除してから、正しいテンプレートを使って、Call Concentrator タイプの新しいアプリケーション オブジェクトを作成します。ICON アプリケーション オブジェクトの作成と構成の詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のインストールと構成についての章を参照してください。

- ICON アプリケーション オブジェクトの [ 接続 ] タブに、Database Access Point (DAP) アプリケーション オブジェクトの割り当てがない。

**解決方法 :** ICON アプリケーション オブジェクトの [ 接続 ] タブに、DAP アプリケーション オブジェクトを追加します。このオブジェクトを介して、ICON インスタンスが Interaction Database (IDB) にアクセスします。

- ICON アプリケーション オブジェクトの [ 接続 ] タブに割り当てられた DAP アプリケーション オブジェクトに、DB Server アプリケーションが関連付けられていない。

**解決方法 :** DB Server を DAP アプリケーション オブジェクトに関連付けます。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のインストールと構成についての章を参照してください。

- すべての添付データを処理するように ICON インスタンスが構成されているが (role = gud)、adata-spec-name 構成オプションで指定されたファイルを ICON が開くことができない。以下に示すログ ファイルのエラーメッセージが、この条件が存在することを示している。

Std 02016 Unable to open attached data file '<attached data specification file name>', error code XXX (Std 02016 添付データ ファイル '< 添付データ仕様ファイル名 >' を開けません。エラーコード XXX)

**解決方法 :** 以下の項目を確認し、必要ならば修正します。

- ◆ adata-spec-name 構成オプションで指定されたファイルが存在するか。ファイルが存在しない場合は、新規ファイルを作成するか、Interaction Concentrator インストールパッケージにあるデフォルトの添付データ仕様ファイル (ccon\_adata\_spec.xml) を使用する。
- ◆ Interaction Concentrator ユーザ (ICON を起動したアカウント) が、添付データ仕様ファイルを読み取る権限を持っているか。
- 永続キュー ファイルが破損している。

**解決方法:** 以下のいずれかの手順で、ICON で強制的に新規永続キュー ファイルを作成します。

- ◆ オペレーティング システム コマンドを使用して、破損した .pq ファイルを移動するか、名前を変更する。再起動すると、新しい .pq ファイルが元のファイル名で元の場所に作成される。
- ◆ ICON アプリケーション オブジェクトの pq-dbname 構成オプションをリセットする。再起動すると、新しい .pq ファイルが新しいファイル名で指定した場所に作成される。pq-dbname 構成オプションの詳細については、『Interaction Concentrator 7.6 Deployment Guide』の構成についての章を参照。

いずれの場合も、古い .pq ファイルの未処理のデータは、ICON と IDB が使用できなくなります。

- apstorage.db ファイルが常駐するディスクに空き領域がない。

**解決方法 :** ディスク領域を解放するか、ディスク領域を追加します。apstorage.db ファイルの詳細については、[122 ページ](#)の「エージェント固有データの HA」を参照してください。

---

## ランタイムの問題

以下にランタイムに関するよくある問題を示します。

- ICON が T-Server または Interaction Server に接続できない ([166 ページ](#)の「T-Server または Interaction Server に接続できない」を参照)。
- ICON が T-Server から呼関連イベントを受信できない ([166 ページ](#)の「ICON が T-Server から呼関連イベントを受信できない」を参照)。
- ICON がデータベースの情報を書き込むことができない ([167 ページ](#)の「ICON がデータベースに情報を書き込むことができない」を参照)。
- ICON と Configuration Database の同期が失われた ([168 ページ](#)の「ICON と Configuration Database の同期が失われる」を参照)。

## T-Server または Interaction Server に接続できない

以下のような原因が考えられます。

- ICONアプリケーション オブジェクトの [接続] タブに、T-Serverアプリケーション オブジェクトまたは Interaction Server アプリケーション オブジェクトの割り当てがない。

**解決方法:** ICONアプリケーション オブジェクトの [接続] タブに、T-Server または Interaction Server アプリケーション オブジェクトを追加します。ここから、ICON インスタンスがインタラクション関連の情報を受け取ります。

- T-Server または Interaction Server アプリケーションが実行されていない、またはネットワークを介してアプリケーションに接続できない。

**解決方法:** アプリケーションを起動するか、ネットワーク接続を再度確立します。

- T-Server または Interaction Server アプリケーション オブジェクトが、交換機リンクに接続できない。

**解決方法:** ご使用の T-Server または Multimedia Interaction Server について、該当するトラブルシューティング ガイドを参照してください。

- T-Server または Interaction Server アプリケーション オブジェクトのリリースが Interaction Concentrator と互換性がない。Interaction Concentrator の任意のリリースに必要な最小バージョンは、T-Server リリース 7.2 で、Interaction Concentrator がマルチメディアをサポートするために必要な最小バージョンは、Multimedia Interaction Server リリース 7.5。Interaction Concentrator と他の Genesys コンポーネントの互換性と相互運用性の詳細については、『Genesys 7 Migration Guide』の Interaction Concentrator の項の最初の章を参照。

**解決方法:** T-Server または Interaction Server アプリケーション オブジェクトを互換性のあるリリースにアップグレードします。

- T-Server アプリケーション オブジェクトに関連付けられた交換機オブジェクトに、必要なすべての DN オブジェクトが構成されていない。

**解決方法:** DN オブジェクトを作成します。詳細については、ご使用の T-Server の『Deployment Guide』を参照してください。

## ICON が T-Server から呼関連イベントを受信できない

以下のような原因が考えられます。

- ICONアプリケーション オブジェクトの [接続] タブで変更を行った後で、ICON を再起動しなかった。

**解決方法:** ICON を停止して、再起動します。詳細については、[97 ページ](#)の「Interaction Concentrator の起動と停止」を参照してください。

- ICON アプリケーション オブジェクトと T-Server の間に接続がない。[166 ページ](#)の「T-Server または Interaction Server に接続できない」を参照。

## ICON がデータベースに情報を書き込むことができない

以下のような原因が考えられます。

- DAP アプリケーション オブジェクトで指定したデータベース パラメータが正しくない (ユーザ名とパスワードなど)。

**解決方法 :** DAP アプリケーション オブジェクトの [DB 情報] タブで正しい値を指定してから、ICON を再起動します。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のインストールと構成についての章を参照してください。

- DB Server が実行されていない、またはネットワークを介してアクセスできない。

**解決方法 :** DB Server を起動するか、ネットワーク接続を再度確立します。

- RDBMS サーバを使用できない、または DB Server が接続しようとする IDB を使用できない。

**解決方法 :** データベース サーバとデータベースを使用可能にするために、必要な手順を行います。

- DAP アプリケーション オブジェクトに設定されている役割が原因で、特定のクラスの情報をデータベースに書き込めない。

**解決方法 :** DAP アプリケーション オブジェクトの role オプションの設定を変更します。ICON を再起動します。DAP の設定の詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のインストールと構成についての章を参照してください。role オプションの詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』の構成オプションについての章を参照してください。

- Interaction Concentrator の初期化スクリプトによって、IDB が初期化されていない。

**解決方法 :** Interaction Concentrator の初期化スクリプトを実行します。詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』のインストールと構成についての章を参照してください。

- ICON アプリケーション オブジェクトの [接続] タブで変更を行った後で、ICON を再起動しなかった。

**解決方法 :** ICON を停止して、再起動します。詳細については、[97 ページ](#)の「Interaction Concentrator の起動と停止」を参照してください。

- レコードがインメモリ キューに蓄積され、IDB に書き込まれない。

**解決方法 :** これは問題ではない可能性があります。インメモリ キューから永続キューへのレコードの転送をサイズのしきい値でトリガするかタイムアウトでトリガするかは、構成オプションによって制御されます。レコードは永続キューから IDB に書き込まれます。転送をトリガするイベントが発生するまで待つから、必要に応じて構成を見直してください。インメモリ キューの構成オプションの詳細については、『*Interaction Concentrator 7.6 Deployment Guide*』の構成オプションについての章を参照してください。

- RDBMS の設定が誤っているため、プログラム ロジックが常にエラーを生成する。たとえば、RDBMS のデータ格納用の空き領域が足りない、ロールバック セグメントが小さすぎるなど。

**解決方法 :** ICON ログ ファイルでエラー メッセージを確認します。HTTP Listener が構成されている場合は、Database Writer パフォーマンスカウンタの Web ページでエラー メッセージを確認することもできます (詳細については、[107 ページ](#)の「Interaction Concentrator のモニタ」を参照)。RDBMS 側で適切な修正を行います。たとえば、エラー メッセージにデータ格納のための空き領域が足りないと記述されている場合は、テーブル スペースを増やします。

エラーが全面的に RDBMS の問題に関連している場合は、ICON の再起動や永続キュー (.pq ファイル) の操作を行う必要はありません。ただし、.pq ファイルが破損し、プログラム ロジックにさらにエラーが発生する場合は、.pq ファイルを置き換える必要があります ([165 ページ](#)の破損した永続キュー ファイルの問題を参照)。

## ICON と Configuration Database の同期が失われる

いくつかの原因で、特にICONのシャットダウン後に、ICONと Configuration Databaseとの同期が失われることがあります。

同期が失われると、IDB に以下のような影響があります。

- ICON が停止している間に作成された構成オブジェクトについてのデータを収集できない。
- ICON が停止している間に該当する構成オブジェクトが削除された場合、構成データに削除のマークを付けることができない。
- ICONがオブジェクト間の関連付けの変更を収集できない(ICONの停止中)。

**解決方法 :** IDB の構成データが Configuration Database のデータと同期していない可能性がある場合は、手動で再同期を行います。詳細については、[135 ページ](#)の「構成データの再同期の方法」を参照してください。

---

## マージ プロシージャの問題

マージ プロシージャ (gsysIRMerge または gsysIRMerge2) の実行中に発生する一般的な問題には、以下のようなものがあります。

- プロシージャが完了しない ([「マージ プロシージャが正常に完了しない」](#)を参照)。
- プロシージャを実行できない ([170 ページ](#)の「マージ プロシージャを実行できない」を参照)。
- マージ プロシージャの実行に時間がかかる、または頻繁に停止する ([171 ページ](#)の「マージ プロシージャの実行に時間がかかるまたは安定しない」を参照)。

マージ プロシージャの詳細については、[141 ページ](#)の「マージ ストアド プロシージャ」を参照してください。

## マージ プロシージャが正常に完了しない

この問題にはいくつかの原因が考えられますが、この項では詳しくは取り上げません。

マージ プロシージャが完了状態まで進行しない場合は、マージ プロシージャの問題の特定の原因を解決するだけでなく、場合によっては、マージ プロシージャが問題から回復するようにプロシージャをリセットする必要もあります（「[マージ プロシージャの回復](#)」を参照）。

---

**注：** 状況によっては、マージ プロシージャの回復は必要ありません。たとえば、デッドロックが原因で、マージ プロシージャが正常に完了しなかった場合などです。この場合は、マージ プロシージャを再実行するだけで、特別なアクションは必要ありません。ただし、マージ プロシージャ内にエラーが見つかった場合、または ICON の複数のインスタンスを同時に実行しているときに、プロシージャの状態が破損した場合は、マージ プロシージャの回復を実行する必要があります。

---

以下のテーブルに、マージ プロシージャの状態に関する情報が格納されます。

- GSYS\_PENDING\_IR
- GSYS\_PENDING\_LINK
- GSYS\_SYSPROCINFO

---

### 手順：

#### マージ プロシージャの回復

**目的：** 問題から回復するためにマージ プロシージャをリセットするには、以下の手順を行います。

#### 手順開始

1. マージ プロシージャ(gsysIRMergeまたはgsysIRMerge2)がアクティブでないことを確認します。
2. GSYS\_PENDING\_IR および GSYS\_PENDING\_LINK テーブルを切り捨てます。
3. GSYS\_SYSPROCINFO テーブルで、PROCNAME = 'gsysirmerge' の行を削除します。
4. 必要ならば、データベース ロックをリセットします（[データベース ロックの手動リセット](#)を参照）。

#### 手順終了

## マージ プロシージャを実行できない

以下のような原因が考えられます。

- ストアド プロシージャが正しく呼び出されなかった。

**解決方法:** gsysIRMerge または gsysIRMerge2 を実行する呼び出しの構文を確認し、必要ならば修正します。詳細については、[149 ページ](#)の「マージ プロシージャの実行」を参照してください。

- 前にマージ プロシージャを実行した後 (または実行を試みた後) で、データベース ロックが解放されなかった。

プロセスが同時に実行されてデータが破損することを防ぐため、マージ プロシージャが IDB にロックを設定します。解放されないロックの問題を解決する手順は、gsysIRMerge を呼び出してプロセスを呼び出したか、gsysIRMerge2 を直接呼び出したかによって異なります。

- gsysIRMerge を呼び出した場合 — 最も安全なロック オーバーライド設定 (ロックを無効にしない) を使用して、gsysIRMerge が gsysIRMerge2 を実行する。前のロックが開放されていない場合は、以下の処理を実行した後で、マージ プロシージャが停止する。

- G\_LOG\_MESSAGES テーブルにレコードを作成する (MESSAGE\_ID = 5020 および MESSAGETEXT = 'LOCKED')。

- G\_LOG\_ATTRS テーブルに対応するレコードを作成する。レコードは属性 'PREVCALLER' (前の呼び出し側の名前) および 'PREVAGE' (前の呼び出し側がロックを取得した後の秒数) を持つ。

**解決方法:** ロックを解放してから、再度 gsysIRMerge を呼び出します。ロックを解放する方法については、[データベース ロックの手動リセット](#)を参照してください。

- gsysIRMerge2 を直接呼び出した場合 — 問題の診断に役立つように、gsysIRMerge2 が RESULT 出力パラメータを出力する。RESULT==2 の場合は、データベースのがロックされているため、プロセスの実行が失敗した。

**解決方法:** 再度 gsysIRMerge2 を呼び出します。このとき、OVERRIDE 入力パラメータと該当する他の入力パラメータに適切な値を指定して、ストアド プロシージャの実行の際にデータベース ロックが無効になるようにします。gsysIRMerge2 のパラメータの詳細については、[148 ページ](#)の「gsysIRMerge2 パラメータ」を参照してください。

---

### 手順:

### データベース ロックの手動リセット

**目的:** gsysIRMerge ストアド プロシージャが IDB に設定したロックを手動で解放します。

### 手順開始

1. ストアド プロシージャが実行されていないことを確認します。確認手順は、ご使用の構築と RDBMS のタイプによって異なります。

2. 以下の文を発行します。正確な構文は RDBMS によって異なります。

```
update GSYS_SYSPROCINFO
set TRANSACTION_ID = 0
where DOMAINID = 0
and PRIMARYID = 99
and PROVIDERTAG = 0
and PROCNAME = '<CALLER>'
```

<CALLER>は、ロックに遭遇したプロシージャ呼び出しで gsysIRMerge2 に渡された CALLER パラメータの値です。gsysIRMerge2 が gsysIRMerge によって呼び出された場合は、CALLER は 'singleIDBMerge' です。

3. [ステップ 2](#) で発行した文をコミットします。

### 手順終了

## マージ プロシージャの実行に時間がかかるまたは安定しない

以下のような原因が考えられます。

- DB2 プラットフォームの場合は、特定のデータベース パラメータのデフォルト値を使用すると、多数のデッドロックが発生する。

**解決方法:** LOCKLIST および MAXLOCKS データベース構成パラメータをかなり大きい値に設定します。

または、ALTER TABLE...LOCKSIZE TABLE 文を G\_IR, G\_CALL および G\_IS\_LINK テーブルに対して実行します。

- 大規模構築の Microsoft SQL プラットフォームの場合は、特定のマージ プロシージャ パラメータのデフォルト値が最適でない。

**解決方法:** G\_DB\_PARAMETERS テーブルの step および limit パラメータをかなり大きい値に設定します。詳細については、[147 ページ](#)の「Microsoft SQL を使用する大規模構築の注意点」を参照してください。





# 索引

## A

ACDQ、定義	69
ACW ( エージェント状態 )	72
adata-default-storage オプション	49
adata-extensions-history オプション	49
adata-reasons-history オプション	49
adata-spec-name オプション	49
adata_spec.xml ファイル。添付データ仕様ファイルを参照	
adata-userdata-history オプション	49
AgentRecordUserTypes オプション	92
AgentSessionID	122
AgentUserFields オプション	92
apstorage.db ファイル	122

## B

busy ( エージェント状態 )	72
-------------------	----

## C

Call Concentrator	91
Call Concentrator アプリケーション タイプ	164
CALLID フィールド	120
CAUSE フィールド	
G_VIRTUAL_QUEUE テーブル内	63
CHAINGUID フィールド	121
Configuration Server	
ICON サーバとのインタフェース	21
IDB との同期	21
データ ソースとして	23

## D

Database Writer ページ	108
DB Server	19
DN	
構成データ	38
マルチメディア交換機	38
DND。着信拒否を参照	32

## E

EventData オプション	92
EventUserEvent	91, 92, 93

## F

FSM。有限状態機械を参照

## G

G_CALL_HISTORY テーブル	117, 120
G_CALL_STAT テーブル	117, 120
G_CALL-USERDATA-CUST	117
G_CALL_USERDATA_CUST1 テーブル	117, 120
G_CALL_USERDATA_CUST2 テーブル	120
G_CALL-USERDATA-CUST テーブル	
IDB テーブル	117
G_CALL_USERDATA_CUST テーブル	120
G_CALL_USERDATA_HISTORY テーブル	120
G_CALL_USERDATA テーブル	117, 120
G_CALL テーブル	117, 120
GCX_* テーブル	122
GC_* テーブル	122
G_DB_PARAMETERS テーブル	
更新	148, 159
ページ プロシージャのパラメータ	158
マージ プロシージャ	144
マージ プロシージャのパラメータ	146
G_DB_PARAMETERS テーブルの更新	148, 159
G_IR_HISTORY テーブル	117
IDB テーブル	
G_IR_HISTORY	120
G_IR テーブル	117
G_IS_LINK_HISTORY テーブル	117, 120
G_IS_LINK テーブル	117, 120
GlobalData オプション	92
G_LOGIN_SESSION テーブル	122
gls-max-inactivity オプション	74
GO_CAMPAIGNHISTORY テーブル	121
GO_CAMPAIGN テーブル	121
GO_CAMPPROP_HIST テーブル	121



GO_FIELDHIST	87
GO_METRICS	83
GO_RECORD	87
GO_SEC_FIELDHIST	87
GO_SECURE_FIELDS	87
GOX_Chain_Call	86, 87
GOX_プレフィックス	34
GO_プレフィックス	34
G_PARTY	28, 39, 117, 120
G_PARTY_HISTORY	28, 40, 120
G_PARTY_STAT	28, 40, 117, 120
G_PROV_CONTROL	143
G_PROV_CONTROL	35
G_ROUTE_RESULT	28, 40, 117, 120
GS_AGENT_STAT	32
GS_AGENT_STAT_WM	32
G_SECURE_USERDATA_HISTORY	47
G_SECURE_USERDATA_HISTORY	33, 117
G_SECURE_USERDATA_HITSTORY	120
G_SYNC_CONTROL	35
GSYS_DNPREMOTE_LOCATION	144
GSYS_DNPREMOTELOCATION	35
GSYS_SYSPROCINFO	144
GSYS_SYSPROCINFO	35
G_TIMECODE	35, 162
G_USERDATA_HISTORY	33, 47, 117
G_VIRTUAL_QUEUE	34, 62, 63, 121
GX_SESSION_ENDPOINT	32
IR_SYNC_QUEUE	119
音声に添付されたデータ用	47
カスタム添付データ用	48
顧客による作成、サポートされない	92
時間設定プロシージャ	161
IDB フィールド	
GSYS_EXT_INT1	122
GSYS_EXT_VCH2	118
GSYS_MSEQ	118
Interaction Concentrator	
7.6 リリースの新機能	25
機能	23
基本的アーキテクチャ	19
コンポーネント	19, 20
時刻フォーマット	25
データ ソース	22
メトリクス、呼	30
Interaction Database	
IDB テーブルも参照	
アウトバウンド関連テーブル	34
エージェント状態関連テーブル	31
カスタム状態関連テーブル	32, 92
仮想キュー テーブル	34
共通データ	92
呼関連テーブル	28
顧客が作成したテーブル、サポートされない	92
ストアド プロシージャ	22
説明	22
添付データ関連テーブル	33
パーティ関連テーブル	28
ログインセッション関連テーブル	31
Interaction Server	
およびエージェント状態	70
データ ソースとして	23
Inter-Server Links。IS-Links を参照	
Inter-Site Call Linkage。IS-Links を参照	
IRID フィールド	120
IR_SYNC_QUEUE テーブル	119
IR_SYNC_TIMEOUT	119
IS-Links	
Interaction Concentrator 内	24
タイムアウト	148
滞留	144
マージ プロシージャのタイムアウトパラメータ	146
<b>K</b>	
KVP。キーと値のペアを参照	
<b>L</b>	
LCA。Local Control Agent を参照	
Local Control Agent	21, 102, 103
login ( エージェント状態 )	71
<b>M</b>	
MCR。マルチメディアを参照	
Message Server	
ICON サーバとのインタフェース	22
monitor オプション	67
MultimediaSwitch	38
<b>N</b>	
notready ( エージェント状態 )	71
null ( エージェント状態 )	71
<b>O</b>	
OCS。Outbound Contact Server を参照	
Outbound Contact Server	
接続	82, 88
データ ソースとして	23
レポートの制限	86
<b>R</b>	
ready ( エージェント状態 )	72
report_reasons オプション	67

report\_targets オプション ..... 67

**S**

SCI。Solution Control Interface *を参照*  
 SCS。Solution Control Server *を参照*  
 Solution Control、ICON サーバとのインタフェース .  
 21  
 Solution Control Interface  
 ICON の起動 ..... 99  
 ICON の停止 ..... 103  
 時間設定プロシージャのログ ..... 162  
 パージ プロシージャのログ ..... 156  
 マージ プロシージャ ログ ..... 143  
 Solution Control Server ..... 103  
 STATUS フィールド  
 G\_VIRTUAL\_QUEUE テーブル内 ..... 63  
 store-event-data オプション ..... 92  
 support-dn-type-5 オプション ..... 68  
 svcUpdateDBParameters ストアド プロシージャ  
 ..... 148, 159

**T**

T-Server  
 およびエージェント状態 ..... 70  
 データ ソースとして ..... 23

**U**

UNIX  
 ICON の起動 ..... 100  
 ICON の停止 ..... 104  
 unknown ( エージェント状態 ) ..... 72

**V**

vq-write-mode オプション ..... 67

**W**

Windows  
 ICON の起動 ..... 101  
 ICON の停止 ..... 105  
 Windows Service Control Manager ..... 105  
 Windows のサービス  
 ICON の起動 ..... 102  
 ICON の停止 ..... 105  
 workmode ..... 76, 77  
 workmode パラメータ、および理由コード ..... 70

**X**

XML スキーマ定義  
 解析の制限 ..... 50  
 属性 ..... 53  
 添付データ仕様ファイル *も参照*

添付データ用 ..... 51

**あ**

アウトバウンド データ  
 IDB テーブル ..... 34  
 IDB に格納されるタイプ ..... 82  
 OCS の制限 ..... 86  
 キャンペーン グループ ..... 88  
 構成の要件 ..... 87  
 メトリクス ..... 83  
 後処理 ( エージェント状態 ) ..... 72

**い**

イベント ( マルチメディア レポート プロトコル ) 39  
 インタラクション キュー エンドポイント ..... 38  
 インタラクション ワークビン エンドポイント ..... 38  
 インメモリ キュー  
 処理 ..... 20

**え**

エージェント状態  
 ACW ..... 72  
 busy ..... 72  
 IDB テーブル ..... 31  
 login ..... 71  
 notready ..... 71  
 null ..... 71  
 ready ..... 72  
 T-Server レポート ..... 70  
 unknown ..... 72  
 移行 ..... 70  
 エージェント状態 *も参照*  
 エージェント メトリクス *も参照*  
 およびメディア タイプ ..... 70, 76  
 カスタム ..... 24  
 定義 ..... 70  
 データ ..... 75  
 統計 ..... 76  
 復元 ..... 73  
 ボイス コールの場合 ..... 70  
 保留 ..... 70  
 マルチメディア インタラクションの場合 ..... 70  
 マルチメディア レポート ..... 70  
 モデル ..... 70  
 有限状態機械 ..... 72  
 理由 ..... 70  
 エージェント状態の復元 ..... 73  
 エージェント デスクトップ アプリケーション ..... 92  
 エージェントの作業場所エンドポイント ..... 38  
 エージェント ログイン セッション。ログイン セッ  
 ション *を参照* ..... 74  
 エラー メッセージ

時間設定プロシージャ	161
ページプロシージャ	156
マージプロシージャ	143
エンコード、マルチバイト文字	25
エンドポイント	
インタラクションキュー	38
インタラクションワークビン	38
エージェントの作業場所	38
およびログインセッション	76
カスタム状態	91
マルチメディア	38
ルーティングストラテジ	38

## か

開始メッセージ	
時間設定プロシージャ	161
マージプロシージャ	143
カスタム状態	
IDB テーブル	32, 92
エージェント デスクトップの構成	93
キーと値のペア	92, 93
構成の要件	92
呼/パーティの関連付け	91
滞留	92
定義	91
データ	92
データの送信	93
同時に複数	93
リリース 7.5 の新機能	24
カスタム状態の記録開始	93
カスタム状態の記録停止	93
カスタム ディスパッチャ	48
構成オプション	50
サンプルスクリプト	48
カスタム添付データ	
IDB テーブル	48
カスタム ディスパッチャも参照	
サンプル仕様ファイル	57
処理	48
仮想キュー	
Interaction Concentrator 内	24
接続	67
仮想キュー データ	
IDB テーブル	34
格納	62
構成オプション	67, 68
処理	62
マルチメディア レポート カスタム メッセージ	39
仮想キュー レコード	121
環境テナント	89

## き

キー、顧客による定義	92
キーと値のペア	
カスタム状態の記録	93
カスタム状態の記録開始	93
カスタム状態の記録停止	93
カスタム状態の場合	92, 93
カスタム添付データ用	48
添付データのマップ	49
機能	
永続キュー	21
機能、Interaction Concentrator	23
キュー、インメモリ キュー、永続キュー、仮想キューを参照	
共通データ	
IDB テーブル	92
呼/パーティの関連付け	91
データ	92

## く

グローバル呼 ID	24
-----------	----

## こ

呼 ID、グローバルに一意	24
高可用性	
エージェント固有データの場合	122
構成データの場合	122
考慮点	124
前提条件	119
データ抽出	117
高可用性モデル	116
交換機	
マルチメディア	38
マルチメディア、DN	38
構成	
Outbound Contact の場合	87
エージェント デスクトップ アプリケーション	93
音声の添付データの処理	49
カスタム状態と共通データの場合	92
マルチメディア用	43
構成オプション	
adata-default-storage	49
adata-extensions-history	49
adata-reasons-history	49
adata-spec-name	49
adata-userdata-history	49
gud-cust-disp	48, 50
gud-cust-disp-groups	48, 50
monitor	67
Outbound Contact の場合	89
report_reasons	67
report_targets	67
support-dn-type-5	68

音声に添付されたデータ用	49
カスタム状態の場合	92
カスタム添付データ用	50
構成データ	
DN	38
IDB テーブル	38
作業場所	38
スクリプト オブジェクト	38
構成のチューニング	21, 107
構成の同期	21
顧客による作成	
テーブル、サポートされない	92
フィールド、サポートされない	92
顧客による定義	
キー	92
状態	91
呼データ	
IDB テーブル	28
このマニュアルに関するご意見	15
呼メトリクス	30
コンソール ウィンドウ	
ICON の停止	104, 105

## さ

サーバの切断	123
作業場所オブジェクトの構成データ	38
サポートされる機能	
Interaction Concentrator	23
アウトバウンド キャンペーン データ	24
インタラクション データ	24
エージェント状態のデータ	24
エージェント ログイン セッションのデータ	24
カスタム状態	24
カスタム添付データ	24
仮想キュー	24
構成データ	23
添付データ	24
添付データ、カスタム	24
ネットワーク	24
ネットワーク コール パーキング	24
マルチバイト文字のエンコード	25
マルチメディア インタラクション	24, 40
リアルタイム レポート	24
ログイン セッションのデータ	24
サンプル スクリプト	48

## し

時間間隔レポート	162
時間設定プロシージャ	
gsysInitTimeCode	161
G_TIMECODE テーブル	162
SCI のログ情報	162

エラー メッセージ	161
開始メッセージ	161
関連する IDB テーブル	161
実行	162
終了メッセージ	161
設定不要	162
入力パラメータ	162
時刻フォーマット	25
持続的キャッシュ ファイル	122
字体スタイル	12
実行	
時間設定プロシージャ	162
ページ プロシージャ	160
マージ プロシージャ	149
終了メッセージ	
時間設定プロシージャ	161
ページ プロシージャ	156
マージ プロシージャ	143
出力パラメータ、マージ プロシージャ用	148
章の要約	10
新機能	
7.6 リリース	25

## す

スクリプト オブジェクトの構成データ	38
スクリプト、サンプル(カスタム ディスパッチャ用)	
	48
スケジューリング、マージ プロシージャ	142, 157
ストアド プロシージャ	
gsysInitTimeCode	161
gsysIRMerge	142
gsysIRMerge2	142
IDB スキーマ内	22
svcUpdateDBParameters	148, 159
時間設定	161
ストアド プロシージャのマージ、時間設定プロシージャ、カスタム ディスパッチャも参照	

## せ

セキュリティ、添付データ	47
接続	
Outbound Contact Server	82, 88
Outbound Contact の場合	88
仮想キュー用	67
マルチメディア用	43
設定	
時間設定プロシージャ	162
ページ プロシージャ	158
マージ プロシージャ	146

## そ

属性、XML スキーマ定義	53
ソフトウェア理由	70, 76

**た**

タイムアウト	
IS-Links	148
ページ プロシージャ ロック	159
滞留	
IS-Links	144
カスタム状態	92
滞留レコード	120

**ち**

着信拒否	
履歴	32
チャット インタラクション。マルチメディアを参照	

**て**

データ格納	
セキュリティ	47
マルチメディア インタラクション	39
データ ソース	22
データの送信、カスタム状態の場合	93
データのソース	22
電子メール インタラクション。マルチメディアを参照	
添付データ	
IDB テーブル	33, 47
キーと値のペアのマップ	49
呼固有	46
セキュリティ	47
添付データの処理、添付データ仕様ファイルも参照	
マルチサイト構築における複製	47
マルチメディア	34
履歴	46
添付データ仕様ファイル	
XML スキーマ定義	50
カスタム添付データ用サンプル	57
名前	49
ボイス コール用サンプル	56
マルチメディア用サンプル	59
添付データの処理	
音声	45
カスタム	48
構成	49
マルチメディア	58

**と**

同一日内レポート	24
同期	
Configuration Database	21
統計	
エージェント状態	32, 76
呼	28, 30
パーティ	28

パフォーマンス。パフォーマンス カウンタ ページも参照	
ログイン セッション	32
同時カスタム状態	93
読者、定義	9
トラブルシューティング	
永続キュー	165
パフォーマンス カウンタ ページの使用	107
マージ プロシージャ	168

**な**

永続キュー	
置き換え	165
機能	21
構成との同期	21
トラブルシューティング	165
破損	165

**に**

入力パラメータ	
gsysPurgeLR	153
時間設定プロシージャ	162
マージ プロシージャ	148

**ね**

ネットワーク環境	24
ネットワーク コール パーキング	24
ネットワーク ベースの Contact Solution	24

**は**

ページ ストアド プロシージャ	
G_DB_PARAMETERS テーブルのパラメータ	158
SCI のログ情報	156
エラー メッセージ	156
実行	160
終了メッセージ	156
設定	158
保持期間	153
呼び出し	160
ロック タイムアウト	159
バージョン番号、マニュアル	12
パーティ	
IDB テーブルのデータ	28
ハードウェア理由	70, 76
パフォーマンス カウンタ ページ	
Database Writer ページ	108
アクセス	107
使用	107
情報のタイプ	108
パフォーマンス カウンタ ページへのアクセス	107
パフォーマンス情報。パフォーマンス カウンタ ページを参照	

パフォーマンスのモニタ	21
パフォーマンス モニタ	21
パラメータ	
時間設定プロシージャ	162
マージ プロシージャ	148

## ほ

保持期間、ページ プロシージャの場合	153
保留エージェント状態、定義	70

## ま

マージストアド プロシージャ	24
G_DB_PARAMETERS テーブルのパラメータ	146
gsysIRMerge2 の呼び出し	150
gsysIRMerge の呼び出し	150
IS-Link のタイムアウト パラメータ	146
SCI のログ情報	143
エラー メッセージ	143
開始メッセージ	143
関連テーブル	142
実行	149
終了メッセージ	143
スケジューリング	142, 157
設定	146
説明	141
トラブルシューティング	168
パラメータ	148
パラメータの更新	148, 159
マニュアル	
誤りに関するコメント	15
規則	12
バージョン番号	12
マルチバイト文字のエンコード、サポート	25
マルチメディア	
DN	38
Interaction Concentrator のデータ	24
エージェント状態	70
エージェント状態の理由コード	76
エンドポイント	38, 76
およびエージェント状態	70
交換機	38
サポートされるシナリオ	40
定義	37
データ格納	39
添付データ仕様ファイルのサンプル	59
添付データ テーブル	34
添付データの処理	58
マルチメディア レポート プロトコル イベント	39

## め

メディア タイプ

およびエージェント状態	70, 76
およびログイン セッション	76
タイプ	70
メディア タイプ、定義	70
メトリクス	
アウトバウンド	83
エージェント状態	32, 76
呼	28, 30
パーティ	28

## も

文字エンコード、マルチバイト	25
モデル	
アウトバウンド キャンペーン	80
アウトバウンド チェーン	81
エージェント状態	70
ログイン セッション	74

## ゆ

有限状態機械	
エージェント状態	72
ログイン セッション	75

## り

リアルタイム レポート	24
理由	
エージェント状態	70
ソフトウェア	70, 76
ハードウェア	70, 76
マルチメディア レポート	70
理由コード	
マルチメディアのエージェント状態の場合	76

## る

ルーティング ストラテジ エンドポイント	38
----------------------	----

## ろ

ログイン セッション	
IDB テーブル	31
およびマルチメディア エンドポイント	76
およびメディア タイプ	76
開始	74
終了	75
定義	70
データ	75
モデル	74
有限状態機械	75
ロック タイムアウト、ページ プロシージャ	159