



製品ドキュメント

---

# ER/Studio® Repository

インストールおよび管理ガイド

バージョン 6.0  
2010 年 10 月 発行

---

© 2010 Embarcadero Technologies, Inc. Embarcadero、Embarcadero Technologies のロゴ、およびすべての Embarcadero Technologies の製品名またはサービス名は、Embarcadero Technologies, Inc. の商標または登録商標です。その他の商標名に関する権利はすべてその所有者に帰属します。

エンバカデロ・テクノロジーズは、1993年にデータベース ツール ベンダーとして設立され、2008年にポーランドの開発ツール部門「CodeGear」との合併によって、アプリケーション開発者とデータベース技術者が多様な環境でソフトウェアアプリケーションを設計、構築、実行するためのツールを提供する最大規模の独立系ツール ベンダーとなりました。米国企業の総収入ランキング「フォーチュン100」のうち90以上の企業と、世界で300万以上のコミュニティが、エンバカデロの Delphi®、C++Builder®、JBuilder® といった CodeGear™ 製品や ER/Studio®、DBArtisan®、RapidSQL® をはじめとする DatabaseGear™ 製品を採用し、生産性の向上と革新的なソフトウェア開発を実現しています。エンバカデロ・テクノロジーズは、サンフランシスコに本社を置き、世界各国に支社を展開しています。詳細は、[www.embarcadero.com/jp](http://www.embarcadero.com/jp) をご覧ください。

2010年10月25日

**CORPORATE HEADQUARTERS**  
100 CALIFORNIA STREET  
12TH FLOOR  
SAN FRANCISCO, CALIFORNIA  
94111 USA

**EMEA HEADQUARTERS**  
YORK HOUSE  
18 YORK ROAD  
MAIDENHEAD, BERKSHIRE  
SL6 1SF, UNITED KINGDOM

**ASIA-PACIFIC HEADQUARTERS**  
L7, 313 LA TROBE STREET  
MELBOURNE VIC 3000  
AUSTRALIA

# 目次

ER/Studio Repository へようこそ	5
ER/Studio Repository 6.0 の新機能	5
重要なお知らせ：製品名の変更	6
このドキュメントについて	6
製品の追加情報	6
ER/Studio Repository のインストール	7
ER/Studio Repository インストールのチェックリスト	8
システム要件	9
Database Server の要件	10
Repository Database のデータ増加率を検討する	11
ER/Studio Repository のインストールを準備する	12
IBM DB2 for LUW のユーザー一時表スペースを作成する	12
IBM DB2 for LUW のデータベースをチューニングする	13
Repository をインストールする	13
Repository に接続する	14
ER/Studio Repository をアンインストールする	15
ER/Studio Repository の管理	17
Repository を理解および管理する	17
Repository のアーキテクチャと設計	18
Repository クライアント	19
Repository Server	19
Repository ファイルを管理する	19
Repository Database	20
Repository および Portal を構成する	21
Repository のパフォーマンスを最適化する	24
トラブル発生に備える	26
Repository でレポートを実行する	26
Repository をバックアップおよび復元する	26
Repository にクエリーを実行する	26
Repository のセキュリティを確保する	27
ロールを作成および管理する	30
ユーザーを作成および管理する	36
Repository 項目へのユーザー アクセスを付与および禁止する	38



# ER/Studio Repository へようこそ

ER/Studio Repository は、サーバーサイドのモデル管理システムです。モデルの共同開発、バージョン管理、セキュリティの確保、およびモデル要素の再利用など、チーム環境におけるモデリング作業の課題を解決します。ER/Studio Repository を使用すると、データおよびビジネスプロセスのモデリングプロジェクトに共同で取り組むユーザーの生産性を高めることができ、リアルタイムの同時アクセスによって、チームメンバーは、プロジェクト全体の資産を共有および再利用することができます。

- モデルの共同開発時にモデルとオブジェクトへの同時アクセスが可能
- モデルとオブジェクトのバージョン管理
- エンタープライズデータディクショナリの標準化
- リアルタイムのオブジェクトステータス通知機能

## ER/Studio Repository 6.0 の新機能

### ER/Studio Repository のパフォーマンス向上

ER/Studio Repository ソフトウェアがマルチスレッド化され、一部の処理を同時に実行できるようになりました。Repository のマルチスレッド処理に関する重要なルールを以下に説明します。

- 特定のダイアグラム/データディクショナリで、チェックインのトランザクションが処理中の場合、そのダイアグラム/データディクショナリに対する他のすべての操作は、チェックインの完了後に処理されます。
- 追加またはチェックインのトランザクションが処理中の場合、その他の追加またはチェックインのトランザクションは、最初のトランザクションの完了後に処理されます。このルールは、名前付きリリースの設定、分岐、および分岐/マージのチェックイン操作にも適用されます。
- その他の一般的なタスクのほとんど（ログイン、ログアウト、取得、チェックアウト、およびセキュリティセンターの操作など）は、サーバーによって同時に処理されます。ただし、データベースレベルのロック機構に影響を受けることがあります。

## 重要なお知らせ：製品名の変更

ER/Studio ファミリーで変更された製品名の一覧を次に示します。

古い製品名	新しい製品名	新しい製品名の省略形
ER/Studio	ER/Studio Data Architect	ER/Studio DA
ER/Studio Repository	ER/Studio Repository	Repository
ER/Studio Portal	ER/Studio Portal	Portal
EA/Studio	ER/Studio Business Architect	ER/Studio BA

## このドキュメントについて

『ER/Studio Repository インストールおよび管理ガイド』は、Repository のメインリファレンスです。Repository のインストールと管理についての情報が記載されています。HTML ヘルプ形式で、ER/Studio DA のオンラインヘルプとして利用できます。また、印刷用に PDF 形式でも提供されています。

このドキュメントに記載されていない追加情報にアクセスするには、「[製品の追加情報](#)」を参照してください。

## 製品の追加情報

エンバカデロ・テクノロジーズの Web サイトでは、製品に関する技術資料、FAQ、サンプルマクロなどを公開しています。

[www.embarcadero.com/jp/support](http://www.embarcadero.com/jp/support) に移動するか、または次のリンクをクリックして参照してください。

- [関連ドキュメント](#)
- [オンラインデモ](#)
- [技術文書](#)
- [ディスカッションフォーラム](#)

# ER/Studio Repository のインストール

サーバーサイドの Repository を使用すると、チームワークと企業レベルでの共同作業を強化できます。ER/Studio Repository は、ER/Studio の Enterprise Edition に含まれています。ER/Studio Repository は、2つのコンポーネントから構成されます。

- **Repository Server:** ER/Studio Data Architect と Repository Database の間を結びつけます。このサーバーでは、データベースにクエリーを実行してデータを取得したり、データベースにデータを挿入するための SQL コードが作成されます。Repository Server は、次のコンポーネントから構成されます。
  - Repository Communication Server
  - Repository Event and Dispatch Server
  - Repository Database Server
- **Repository Database:** Repository のダイアグラムとそのオブジェクトに関する情報を格納します。

このセクションには次のトピックが含まれています。

- [ER/Studio Repository インストールのチェックリスト](#) (8 ページ)
- [システム要件](#) (9 ページ)
- [Database Server の要件](#) (10 ページ)
- [Repository Database のデータ増加率を検討する](#) (11 ページ)
- [ER/Studio Repository のインストールを準備する](#) (12 ページ)
- [Repository をインストールする](#) (13 ページ)
- [Repository に接続する](#) (14 ページ)
- [ER/Studio Repository をアンインストールする](#) (15 ページ)

## ER/Studio Repository インストールのチェックリスト

ER/Studio Repository を正しくインストールおよび構成できるように、次のチェックリストを使用してください。

完了	N/A	要件
		Repository が含まれている ER/Studio Enterprise をダウンロードする。 14 日間の試用が可能な ER/Studio Enterprise を次の Web サイトからダウンロードできます。 後からライセンスを取得することもできます。 <a href="http://www.embarcadero.com/jp/downloads">www.embarcadero.com/jp/downloads</a>
		ER/Studio の ReadMe を確認する。 インストールおよび使用方法に関する最新情報については、次の Web サイトにある ReadMe を参照してください。 <a href="http://docs.embarcadero.com">docs.embarcadero.com</a>
		システムの互換性を確認する。 9 ページの「 <a href="#">システム要件</a> 」を参照してください。
		Repository Database を最適化する。 Repository のパフォーマンスを最適化する方法については、12 ページの「 <a href="#">ER/Studio Repository のインストールを準備する</a> 」を参照してください。
		Repository のデータ増加率を検討する。 Repository Database に必要なサイズを見積もるには、11 ページの「 <a href="#">Repository Database のデータ増加率を検討する</a> 」を参照してください。
		ER/Studio Repository をインストールする。 Repository のインストール手順については、13 ページの「 <a href="#">Repository をインストールする</a> 」を参照してください。
		Repository に接続する。 ER/Studio Data Architect から Repository に接続する手順については、14 ページの「 <a href="#">Repository に接続する</a> 」を参照してください。
		ER/Studio Data Architect を起動して Repository の操作をテストする。 ER/Studio Data Architect の使用方法については、『ER/Studio Data Architect Evaluation Guide』を参照してください。次の Web サイトからダウンロードできます。 <a href="http://docs.embarcadero.com">docs.embarcadero.com</a>



## システム要件

ER/Studio Repository をインストールする前に、以下の要件を確認してください。ER/Studio Repository で最適なパフォーマンスを得るには、これらの要件を満たすことが必要です。

	ER/Studio Repository の要件	
ハードウェア	Repository Database Server	Repository Server
		使用するデータベース プラットフォームの要件に準拠
プロセッサ		Pentium 4 以降
RAM		2GB
ディスク容量		50MB
DVD ドライブ		はい
オペレーティングシステム		次の OS をサポート： <ul style="list-style-type: none"> <li>• Windows Server 2008 (32 ビットおよび 64 ビットモード)</li> <li>• Windows Vista</li> <li>• Windows XP</li> <li>• Windows 2000</li> <li>• Windows 2003 (32 ビットモード)</li> </ul>
ソフトウェア	TCP (Transaction Control Protocol) が動作している 西ヨーロッパ言語の文字セット (UNICODE および UTF8 はサポートしていません)	TCP が動作している
	サポートされる DBMS (10 ページの「 <a href="#">Database Server の要件</a> 」を参照してください。)	
権限	サーバー上にオブジェクトを作成できる DBMS 権限	サーバー上にオブジェクトを作成できる DBMS 権限
	ローカルの管理者権限	ローカルの管理者権限

**メモ：** このハードウェア要件では、5～10 ユーザーの環境に ER/Studio Repository をインストールすることを想定しています。10 ユーザーを超える環境では、ユーザーの増加に伴うトランザクション数の増加に対応するために、上記の要件を上回るハードウェアをご使用ください。

## Database Server の要件

次の表に、Repository がサポートするデータベースの一覧を示します。Repository からデータベースに接続するには、Repository を実行中のマシンに、対応する DBMS クライアントユーティリティをインストールする必要があります。ER/Studio Data Architect から、これらのデータベースに対してリバースエンジニアリングを実行する場合、ER/Studio Data Architect を実行中のマシンにも DBMS クライアントユーティリティをインストールする必要があります。

ER/Studio Repository Database Server の要件	
Repository でサポートする DBMS	対応する DBMS クライアントユーティリティ ソフトウェア
IBM DB2 LUW 7.x *, 8.x、および 9.x Server	対応するバージョンの IBM DB2 UDB クライアントユーティリティ (7.x、8.x、または 9.x)
Microsoft SQL Server 2000、2005、および 2008	対応するバージョンの SQL Server Client Access (2000、2005、または 2008)
Microsoft SQL Server 7.0	Microsoft Data Access Component (MDAC) 2.6 以降
Oracle 8.1.x、9.0.x、9.2.0.x、10g、および 11g	対応するバージョンの Oracle クライアントユーティリティ (8.1.7.4、9.0、9.2.0、10g、または 11g)
Sybase ASE 12.x または 15.x	Sybase 11.9 以降のクライアントユーティリティ

### 重要な注意点

- ER/Studio Repository は、32 ビットバージョンのデータベースクライアントソフトウェアを使用して、上記のデータベースプラットフォームと動作することが保証されています。64 ビットクライアントはサポートしていません。
- ER/Studio Repository Server と ER/Studio Repository Database を同じマシンにインストールする場合は、先に示したハードウェア要件を上回っていることを確認してください。
- ER/Studio Repository 6.x は、ER/Studio Data Architect 9.x 以降と使用する必要があります。
- ER/Studio Repository のアップグレード後に ER/Studio Portal を使用するには、ER/Studio Portal も最新バージョンにアップグレードしてください。
- DBMS クライアントユーティリティソフトウェアのインストールは、Repository Server のインストール前に完了している必要があります。
- Microsoft SQL Server で、大文字と小文字を区別する命名法を使用している場合、ER/Studio Repository から接続エラーメッセージが表示されることがあります。この問題を回避するには、大文字と小文字を区別しない文字セット (Latin\_General\_CI\_AS など) を使用して空のデータベースを作成してから、Repository Database メンテナンスユーティリティでデータベースを初期化します。
- IBM DB2 UDB 7.x サーバーでは、ストアードプロシージャが正しく動作するために、DB2 インスタンスが存在するマシンに C++ コンパイラが必要です。

- すべてのバージョンの IBM DB2 UDB サーバーでは、Repository で最適なパフォーマンスを得るために、ユーザー一時表スペースを作成してデータベースをチューニングする必要があります。これによって、メモリやログ容量の不足を回避できます。詳細については、12 ページの「[ER/Studio Repository のインストールを準備する](#)」を参照してください。
- すべてのバージョンの IBM DB2 UDB サーバーでは、Repository Database へのクライアント接続が、システムの ODBC データ ソースとして登録されている必要があります。データ ソース名は、クライアント接続で指定されたデータベース エイリアスと一致している必要があります。

## Repository Database のデータ増加率を検討する

プラットフォームの種類によらず、少なくとも 300MB の容量をデータベースに割り当てます。また、自動拡張を設定して、モデルのサイズと数が増加してもデータベースが対応できるようにしてください。さらに、データベースのログ ファイルに 50MB の容量を割り当てます。

以下の表に、いくつかの Repository Database について、ダイアグラムを追加する前の初期サイズと、サンプルダイアグラム（小、中、大）に対するデータベースサイズの概算値を示します。

RDBMS	初期サイズ (MB)	小さいダイアグラム (MB)	中サイズのダイアグラム (MB)	大きなダイアグラム (MB)
Oracle 8	6	0.5	14	64
Microsoft SQL Server 7	7.8	0.7	14.7	40.9
Sybase ASE 12	3.9	0.8	28.9	89.1
IBM DB2 UDB	3.2	0.8	14.6	50

以下の表に、各サンプルダイアグラム（小、中、大）の特徴を示します。

サンプルダイアグラム	エンティティ	属性	ビュー	リレーションシップ	DM1 ファイルのサイズ (KB)
小	22	125	2	20	184
中	302	2516	32	606	2153
大	722	10629	0	1542	6967

Repository を正しくインストールするには、次のトピックを参照してください。

- [ER/Studio Repository のインストールを準備する](#) (12 ページ)
- [Repository をインストールする](#) (13 ページ)

**メモ：** ER/Studio Repository 6.x をインストールする場合、ER/Studio Data Architect 9.x 以降をインストールしてください。旧バージョンの ER/Studio Data Architect は、ER/Studio Repository 6.x と互換性がありません。

## ER/Studio Repository のインストールを準備する

### Microsoft SQL Server

Microsoft SQL Server で、大文字と小文字を区別する命名法を使用している場合、ER/Studio Repository から接続エラーメッセージが表示されることがあります。この問題を回避するには、大文字と小文字を区別しない文字セット (Latin1\_General\_CI\_AS など) を使用して空のデータベースを作成してから、Repository のユーティリティでデータベースを初期化します。また、サーバーのセキュリティには混合認証を使用してください。

### IBM DB2 for LUW

ER/Studio Repository のデータベースが IBM DB2 for LUW に保存される場合、Repository のパフォーマンスを最適化するために次の手順を実行してください。

- [IBM DB2 for LUW のユーザー一時表スペースを作成する](#) (12 ページ)
- [IBM DB2 for LUW のデータベースをチューニングする](#) (13 ページ)

### IBM DB2 for LUW のユーザー一時表スペースを作成する

DB2 UDB データベースでは、一時テーブルを作成するために、少なくとも 4KB のページサイズを持つユーザー一時表スペースが必要です。

**ヒント:** システム管理のユーザー一時表スペースを使用してください。システム管理の場合、ユーザー一時表スペースを自動的に拡張できるためです。

**ヒント:** DBArtisan を使用して、システム管理のユーザー一時表スペースを作成することができます。

任意のツールを使用して、ユーザー一時表スペースを作成します。または、次の SQL を使用することもできます。

**メモ:** このスクリプトを実行する前に、コンテナのパスを変更してください。

```
CREATE USER TEMPORARY TABLESPACE REPOTEMP
  PAGESIZE 4096
  MANAGED BY SYSTEM
  USING('D:\DB2\NODE0000\SQL00002\SQLT0004.0')
// here you specify the physical location of the table space
  EXTENTSIZE 32
  PREFETCHSIZE 16
  BUFFERPOOL IBMDEFAULTBP
  OVERHEAD 24.10
  TRANSFERRATE 0.90
;
```

## IBM DB2 for LUW のデータベースをチューニングする

この手順では、ログ容量を使い切らないように DB2 データベースをチューニングします。

1. アプリケーション ヒープ サイズおよびアプリケーション制御ヒープ サイズを、2048×4KB に増やします。

**ヒント:** アプリケーション ヒープ サイズを増やすには、DB2 コントロール センターを開き、データベースを右クリックして [パラメーターの構成] をクリックします。[パフォーマンス] 項目までスクロールして、アプリケーション ヒープ サイズ (applheapsz) およびアプリケーション制御ヒープ サイズ (apptlheapsz) の値を変更します。

2. ログ ファイルの容量を、10,000×4KB (40MB) に増やします。
3. 1 次ログ ファイルと 2 次ログ ファイルの値を、それぞれ「3」および「2」から「10」および「10」に変更します。1 次ログの容量は合計 400MB になり、これは中規模から大規模のダイアグラムでも十分な容量です。

**ヒント:** 1 次ログ ファイルと 2 次ログ ファイルのサイズを増やすには、IBM コントロール センターの [データベース構成] ページで、[ログ] 項目までスクロールして値を変更します。

## Repository をインストールする

**メモ:** 新しい Repository Database を作成するには、DBMS サーバー上のデータベースの管理者権限アカウントが必要です。

1. ローカルの管理者権限を持つアカウントで Windows にログオンします。
2. ER/Studio Repository 6.x のインストール プログラムを起動します。
3. インストール ウィザードの指示に従い、インストールを進めます。

表示されるオプションは分かりやすいものですが、次のオプションは必ず選択してください。

- **新しいデータベースを作成する**

### メモ

インストール時には、テーブル、インデックス、ビュー、およびストアド プロシージャが追加されます。インストールの大部分は説明が不要な手順ですので、画面の指示に従ってください。一部の手順について補足説明します。

- [データベース名] : 既に Repository Server がインストールされた DBMS サーバーに Repository Server をインストールするには、新規サーバー用に固有のデータベース名を入力します。データベース名を変更すると、データ ファイルとログ ファイルの名前にも反映されます。

- **Microsoft SQL Server 7.0 以降での自動拡張**：データベース ファイルの自動拡張を許可するには、[拡張を許可する] をクリックして [拡張率] の値を入力します。拡張率は、データベースが一杯になったときにデータベースに追加する容量を、現在のデータベース サイズに対するパーセント単位か、または MB 単位のいずれかで表します。ファイルの最大サイズは、[無制限] をクリックするか、または [サイズ] をクリックして最大サイズを MB 単位で入力します。
- **[新しい Oracle ユーザー]** ダイアログ ボックスを使用して、新規ユーザーを作成し、そのユーザーに対する一時表領域の情報を指定します。このダイアログ ボックスは分かりやすい内容ですが、ユーザー認証オプションについて補足説明します。
  - [パスワード]：入力されたユーザー名とパスワードを認証します。
  - [外部]：既存のオペレーティング システムのユーザー名に対応するデータベースのユーザー名を認証します。
  - [グローバル]：ユーザーにアクセスを許可するために、セキュリティ ドメインで集中管理されているユーザー名およびパスワードの情報を取得します。このオプションが利用できるのは Oracle 8 のみです。
- **[エラー ログ ビューア]**：Repository Database を作成する際に問題が発生すると、エラー ログ ビューアが表示されます。エラーが発生した場合、ご使用の環境が Repository Database の要件を満たしていることを確認してください。Repository Database 作成の詳細については、エンバカデロ・テクノロジーズの技術サポートにお問い合わせください。

## Repository に接続する

1. ER/Studio Data Architect または ER/Studio Viewer を起動します。
2. [Repository] メニューの [オプション] をクリックします。
3. [Repository オプション] ダイアログ ボックスの [サーバー マシン] フィールドに、Repository サーバー マシンのホスト名または IP アドレスを入力して、[OK] をクリックします。
4. Repository にログインします。

[Repository] メニューの [ログイン] をクリックし、ユーザー名とパスワードを入力して [OK] をクリックします。

これで、ER/Studio Data Architect から ER/Studio Repository を使用する準備が整いました。ER/Studio Repository の詳しい使用方法については、『ER/Studio Data Architect ユーザー ガイド』を参照してください（ER/Studio DA 上で [ヘルプ] メニューの [ユーザー ガイド] をクリックします）。

## ER/Studio Repository をアンインストールする

Windows のコントロール パネルで [プログラムの追加と削除] を開き、[Embarcadero ER/Studio Repository X.X] を選択して [削除] をクリックします。画面の指示に従い、すべての ER/Studio Repository コンポーネントを削除します。

**メモ：** ER/Studio Repository をアンインストールしても、Repository Database は削除されません。データベース管理者が削除する必要があります。





# ER/Studio Repository の管理

このセクションでは、ER/Studio Data Architect (ER/Studio DA) のインストールおよび構成担当者、ER/Studio Repository データベースの管理者向けの情報を説明します。次のトピックが含まれています。

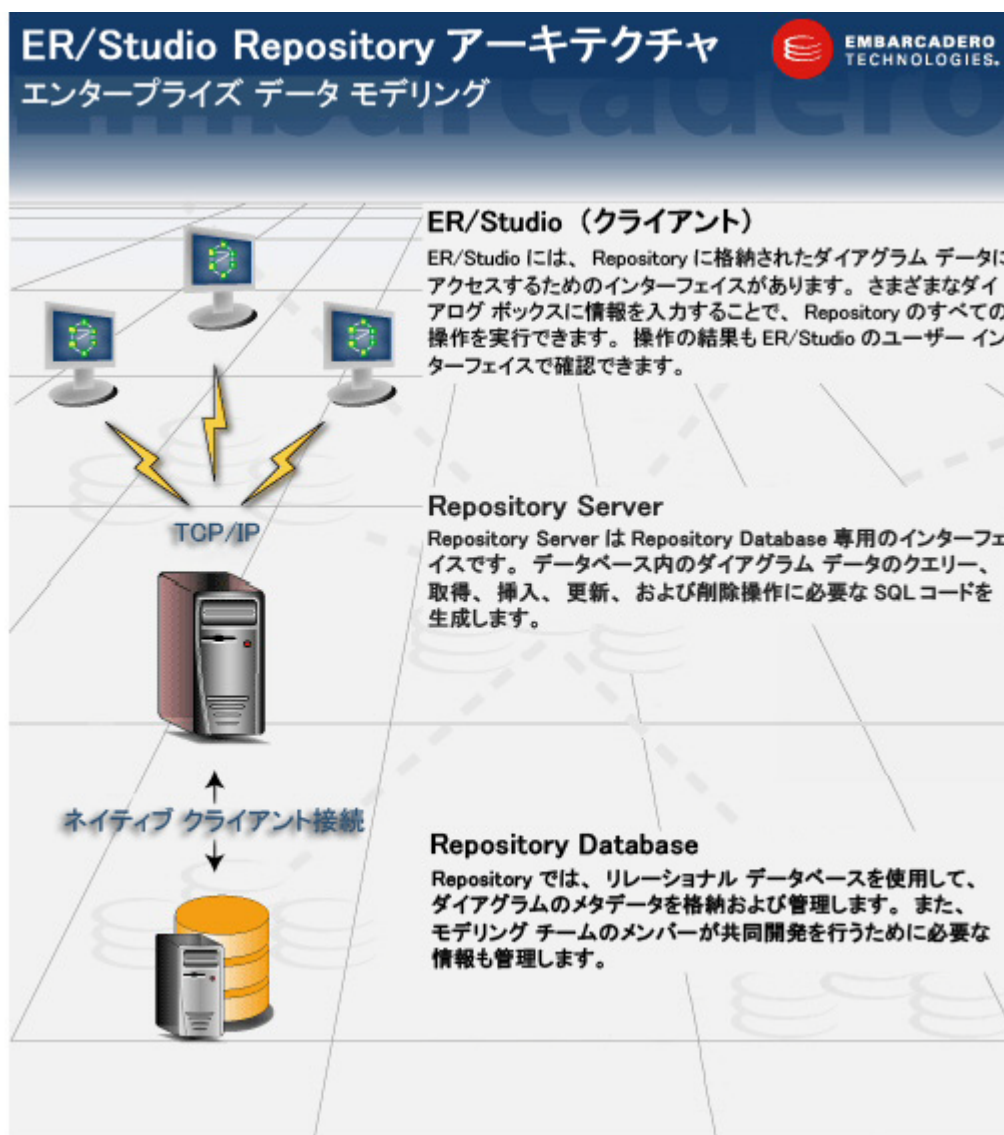
- [Repository を理解および管理する](#) (17 ページ)
- [Repository のセキュリティを確保する](#) (27 ページ)
- [Repository ファイルを管理する](#) (19 ページ)

## Repository を理解および管理する

このセクションには次のトピックが含まれています。

- [Repository のアーキテクチャと設計](#) (18 ページ)
- [Repository および Portal を構成する](#) (21 ページ)
- [トラブル発生に備える](#) (26 ページ)

## Repository のアーキテクチャと設計



Repository では、ネイティブ クライアント接続および TCP/IP プロトコルを使用して、ER/Studio DA、Repository Server、および Repository Database 間の情報を送受信します。クライアントとサーバー間の通信はセキュリティで保護されています。クライアントと Repository では SSL がサポートされており、共通の URL とポートを指定して、クライアントと Repository 間でのセキュアな通信を実現します。

SystemModels ディレクトリには、Repository スキーマのメタデータ ダイアグラムがあります (Repository600.dm1)。このダイアグラムにはエンティティ間のリレーションシップがドキュメントの観点から含まれていますが、パフォーマンスの理由から、実際のデータベースには外部キー制約が含まれていません。このファイルは、Repository に対して実行するクエリーを記述する上で役立つ、移行のつながりを示しています。

ソフトウェアのインストール要件と、Repository でサポートされるデータベースの完全な一覧については、10 ページの「[Database Server の要件](#)」を参照してください。

## Repository クライアント

ER/Studio DA のユーザー インターフェイスから、Repository に保存されたダイアグラム データにアクセスできます。[Repository] メニューから、すべての Repository 機能を実行できます。Repository にアクセスするには、ER/Studio DA を起動する必要があります。

[Repository] タブから、Repository のメタデータにアクセスでき、ER/Studio Portal を通じた検索やレポート作成を実行できます。

## Repository Server

Repository Server は、ER/Studio DA と Repository Database をリンクさせます。Repository Server には、データベースにクエリーを実行してデータを取得したり、データベースにデータを挿入するための SQL コードが含まれています。Repository Server は、データベースのデータを読み取り適切な情報を返す操作や、新しいデータをデータベースに書き込む操作の実行要求を受信すると、これらのコードを実行します。

**メモ：** Repository Server は、データベースにデータを追加したり、データベースからデータを取得する役割を果たします。したがって、各 Repository Database に対して 1 つの Repository Server のみを設定できます。

Repository Server は、Repository データの状態を Repository Database に保存して管理します。また、ER/Studio DA ユーザーによって要求されたモデリングの変更操作を記録します。

Repository Server が要求を受け取ると、次の処理を実行します。

1. 操作に必要な権限がユーザーにあることを確認します。
2. Repository Database のデータを要求する操作の場合、Repository Server はデータベースからデータを取得し、要求の送信元である ER/Studio DA のインスタンスに送信します。
3. Repository Database にデータを保存する操作の場合、Repository Server はデータベースにデータを書き込みます。
4. すべての ER/Studio DA クライアントに Repository の新しい状態を通知して、表示の更新に必要な情報をクライアントに送信します。

## Repository ファイルを管理する

Repository Server をインストールすると、次のような複数のフォルダが作成されます。構成オプションによって異なりますが、特にサーバーで「-deleteoff」オプションが設定されている場合は、以下で説明するように RepoIn フォルダと RepoOut フォルダに保存されている古いファイルを手動で削除することができます。

.pri ファイルまたは .pro ファイルはいつでも削除できます。古い日付のファイルは削除しても問題ありません。通常、ファイルはディレクトリに書き込まれるとすぐに処理されるためです。  
 .out ファイルおよび .err ファイルも同様に削除できます。拡張子が .in のファイルは削除すべきではありません。それらのファイルは、Repository サービスが停止した場合を除き、処理が終わった後にフォルダに残っていることはありません。

フォルダ	説明
Repository	次に示すサブフォルダが格納されます。その他、Repository サービスの 3 つの実行可能ファイル、Repository Server のライセンス アプリケーション、使用許諾契約書のコピー、Repository のインストールとライセンスに関するヘルプ ファイルが含まれています。
Logs	Repository フォルダのサブフォルダです。各 Repository サービスでイベントやトランザクションに関する情報を記録するログ ファイルが保存されます。
RepoIn	ER/Studio DA クライアントから受信した要求ファイルが、サーバーで処理されるまで保存されます。古いファイルが保存されている場合は、次の操作を実行します。 <ol style="list-style-type: none"> <li>1. Windows エクスプローラで、このディレクトリを開きます。</li> <li>2. [更新日時] でファイルをソートして、最も古いファイルが先頭にくるようにします。</li> <li>3. 一番上から、削除しても問題ないと思われる場所までファイルを選択します。今日の日付より前に変更されたファイルをすべて選択することをお勧めします。</li> <li>4. 選択したファイルを削除します。</li> </ol>
RepoOut	Repository Server から ER/Studio DA クライアントに送信されるファイルが、処理されるまで保存されます。

Repository のインストール時には、Repository Server と Repository Database を同じマシンにインストールすることも、それぞれ別のマシンにインストールすることもできます。各コンポーネントをどのマシンにインストールしたかに関わらず、定期的に両方のコンポーネントを完全にバックアップすることをお勧めします。

## Repository Database

Repository はデータベースを使用して、Repository のダイアグラムとそのオブジェクトに関する情報を保存します。データベースは、ユーザーがダイアグラムの要素をチェックアウト、チェックイン、追加、または削除するたびに更新されます。Repository Database には、特定の Repository 操作の履歴や、セキュリティのルールとデータも保存されます。

**注意：** Repository Database は、モデルを複数ユーザーで作業できるように設計されたトランザクション システムであるため、Repository テーブルを直接更新することは避けてください。ER/Studio DA を使用せずに Repository Database を更新すると、データが破損する恐れがあります。Embarcadero\ERStudio Repository\Utilities以下のMasterScriptsおよびSQLScriptsフォルダに保存されている SQL スクリプトは、ER/Studio DA の Repository 機能で使用されるものです。これらのスクリプトを ER/Studio DA を介さずに直接使用するのは、エンバカデロ技術サポートからの依頼があった場合のみに留めてください。

**メモ：** Repository に対してクエリー (SELECT 文) を発行して、レポートを作成できます。この目的に使用できる一般的な SQL スクリプトが、Embarcadero\ERStudio Repository\Utilities\QueryScriptsおよびEmbarcadero\ERStudio Data Architect X.X\SQLCode\Repo\_Queries フォルダに保存されています。ユーザー独自のスクリプトを作成することもできます。その場合は、\Program Files\Embarcadero\ERStudio Data Architect X.X\SystemModels フォルダに保存されているメタモデルを参考にしてください。

## Repository および Portal を構成する

Repository の設定項目で、サーバーの詳細情報、Repository の表示設定、チェックアウトポリシー、および通信設定を指定できます。Repository の動作を制御するダイアログ ボックスは2つあります。[Repository オプション](#)と[Repository のプロパティ](#)です。[Repository オプション] ダイアログ ボックスでは、Portal の設定も定義されています。

### Repository と Portal のオプションを設定する

1. Repository にログインします。
2. [Repository] メニューの [オプション] をクリックします。
3. [Repository オプション] ダイアログ ボックスで必要なオプションを設定したら、[OK] をクリックしてダイアログ ボックスを閉じます。

次に、一部のオプションについて補足説明します。

#### [全般] タブ

Repository Server のすべての設定項目を指定できます。

- [サーバー マシン]：ER/Studio Repository のホスト名が表示されます。ボックスの右端をクリックすると Repository 一覧が表示され、他の Repository を選択できます。[リフレッシュ] をクリックすると Repository 一覧が更新され、ネットワークで見つかったすべての Repository のホスト名が表示されます。
- [リフレッシュ]：クリックすると、ネットワーク上で使用できる Repository Server の一覧が更新されます。

- [アクティブ ファイル ディレクトリ] : Repository からチェックアウトまたは取得したファイルが保存されるディレクトリです。アクティブ ファイル ディレクトリは、ネットワーク上の共有ディレクトリではなく、ローカル マシン上のディレクトリを指定する必要があります。ネットワーク上の共有ディレクトリを指定すると Repository 操作でエラーが発生することがあります。  
**注意 :** アクティブ ファイル ディレクトリには、ローカル マシン上のディレクトリを指定してください。ネットワーク上の共有ディレクトリを指定すると、Repository 操作でエラーが発生することがあります。
- [Repository ステータス アイコンを非表示] : Repository ステータス アイコン (鍵アイコンなど) を、モデル エクスプローラ / ウィンドウ (およびデータ リネージ エクスプローラ / ウィンドウ) で非表示にします。アイコンを表示しないでダイアグラムを印刷する際に役立ちます。
- [Repository ステータス ダイアログを表示] : [Repository 操作のステータス] ダイアログ ボックスの表示 / 非表示を指定します。このダイアログ ボックスには、Repository 操作の各手順の所要時間が表示されます。
- [ビュー SQL の警告] : Repository のマージ操作で、複数のユーザーがビューのデータを修正している場合、チェックイン時に警告が表示されます。表示された SQL 警告画面には、今後メッセージを表示しないようにするオプションがあります。[ビュー SQL の警告] オプションを使用して、メッセージを再度有効にすることができます。
- [Repository イベント ポーリングの有効化] : Repository イベント ポーリングにより、Repository 操作に関する情報が返されます。ダイアグラムまたはオブジェクトの状態が、取得可能な最も新しい情報に更新されます。Repository イベント ポーリングを有効にした場合、ポーリングを実行する時間間隔を指定できます。このオプションはデフォルトで有効になっています。
- [Repository イベント ポーリング間隔] : ローカルのダイアグラムおよびオブジェクトと Repository の状態をシンクロするために、ポーリングを実行する時間間隔 (秒単位) です。
- [全般情報を非表示] : パフォーマンス上の問題がある場合、このオプションを選択して、[Repository からの取得] ダイアログ ボックスの [全般情報] 領域に、Repository ステータス情報を表示しないように設定できます。
- [コメントのシンクロの警告を表示] : コメントを Repository とシンクロする前に、ファイルの変更点を保存するようメッセージが表示されます。このオプションがオフの場合は、シンクロ前にファイルが自動保存されます。
- [ダイアグラム名の表示形式] : [Repository からの取得] ダイアログ ボックスなど、Repository ダイアログに表示されるダイアグラム名の表示形式を指定します。[ダイアグラム名 (ファイル名 .dm1)] を選択すると、「Demo1 (Model.dm1)」のように表示されます。「Demo1」は [ダイアグラム プロパティ] ダイアログ ボックスの [名前] フィールドに入力された文字列です ([ファイル] メニューの [ダイアグラム プロパティ] をクリックしてアクセスします)。

## [チェックアウト ポリシー] タブ

チェックアウト ポリシーのプロパティを変更できます。たとえば、チェックアウトしていない Repository 項目を編集しようとしたときに確認メッセージを表示するように指定できます。

- [ER オブジェクト] 領域：モデル オブジェクトの名前や定義など、中核となるデータを Repository へ保存する方法を指定します。
- [レイアウト オブジェクト] 領域：オブジェクトのサイズ、色、配置など、エンティティ オブジェクトの表示情報を Repository へ保存する方法を指定します。

[即時チェックアウト]：[即時チェックアウト] ポリシーを使用するには、サーバー接続が確立していることと、ユーザーがログインしている必要があります。チェックアウトしていないオブジェクトを編集しようとする、メッセージが表示され、オブジェクトをチェックアウトするか、またはオブジェクトをビュー モード（読み取り専用）のエディタで開くかを選択できます。

[遅延チェックアウト]：[遅延チェックアウト] および [確認メッセージを表示] を選択すると、チェックアウトしていないオブジェクトを編集しようとしたときに、そのオブジェクトのチェックアウトを促す確認ダイアログ ボックスが表示されます。確認ダイアログ ボックスで [はい] をクリックするか、または [確認メッセージを表示] が選択されていなかった場合は、そのオブジェクトにアンロック状態の赤いアイコンが付きます。これは、チェックアウトがローカルで実行されたことを示します。（アンロック状態の緑のアイコンは、オブジェクトが Repository からチェックアウトされたことを示し、すべての Repository ユーザーがその状態を確認できます。）このオブジェクトがチェックアウト状態にあることは、Repository には通知されません。実際のチェックアウトが実行されるのは、[Repository] メニューの [ダイアグラム] をポイントし、[遅延チェックアウトの実行] をクリックしたときです。その時点で、他のユーザーによる変更点がチェックインされていた場合、矛盾点があれば解決する必要があります。

## [ER/Studio Portal] タブ

[URL]：ER/Studio Portal の接続情報を指定します。ER/Studio Portal を使用すると、[Repository] タブから Repository のコンテンツを参照したり、Repository コンテンツのレポートを表示することができます。Portal を実行しているサーバーの名前（たとえば datojt090）、またはポータル ログインの URL（たとえば <http://datojt090/ersportal>）を指定します。

[Portal ステータス ダイアログの表示]：Portal 操作中に実行された各手順の所要時間が表示されます。このオプションをオフにすると、Portal 操作中にエラーが発生した場合、[Portal エラー情報] ページが表示されます。

[操作のタイムアウト値]：失敗した Portal 操作がタイムアウトでエラーになるまでの時間を秒単位で指定します。

## メモ

- ファイルを Repository に追加またはチェックインするときは、アクティブ ファイル ディレクトリにファイルを置いてから操作を実行してください。
- Repository Server の設定を構成するときは、インストール時の構成を考慮してください。
- Repository Database と Repository Server は、同じコンピュータに配置する必要はありません。Repository Server の構成は、Repository Database に影響しません。

## Repository のプロパティ

1. Repository のプロパティを読み取り専用で表示するには、Repository にログインします。  
Repository Server のポートを変更するには、Repository からログアウトします。
2. [Repository] メニューの [Repository プロパティ] をクリックします。
3. [サーバー] タブをクリックすると、サーバーの設定が表示されます。[ユーザー] タブをクリックすると、現在のユーザーのログイン名が表示されます。
4. 必要があれば、Repository にログインしていないときに、Repository との通信に使用するポートを変更できます。[サーバー] タブで [ポートの編集] をクリックして、ポートの設定を変更します。[変更の適用] をクリックしてから [閉じる] をクリックし、エディタを閉じます。

[SSL ポート] : バージョン 8.5 から、SSL (Secure Sockets Layer) 通信プロトコルをサポートしました。クライアントと Repository との間のメッセージを暗号化することができます。SSL を使用して ER/Studio DA のネットワーク メッセージを暗号化するには、[Repository に SSL 接続する] チェック ボックスをオンにしてください。

## Repository のパフォーマンスを最適化する

Repository のパフォーマンスを最適化する際に役立つポイントを説明します。

- **専用サーバーを使用する** : Repository のパフォーマンスに直接影響する要素は、ダイアグラムの数とサイズ、およびクライアントと Repository との間で転送されるデータ量です。Repository は専用サーバーで使用することをお勧めします。ただし、業務用のハードウェア (複数のアプリケーションを同時に管理できるマルチプロセッサのラック サーバーなど) を使用している場合はその限りではありません。
- **Repository Server と Repository Database を別々のマシンに配置する** : Repository Server と Repository Database を同じマシンに配置する必要はありません。別々のマシンに配置すると競合が回避されるため、待ち時間が軽減される場合があります。Repository Server と Repository Database を別々のマシンに配置する場合、パフォーマンスを最適化するには、ユーザーと物理的に同じ場所にサーバーを設置します。



- **Repository のフォルダをクリアする**：Repository Server マシンには、  
\\...\\Embarcadero\\ERStudio Repository\\フォルダ以下にRepoInおよびRepoOutフォルダがあります。ネットワーク上の問題により、これらのフォルダ内に複数の項目が作成される場合があります。これらのフォルダに多数のファイルが含まれていると、Repository のパフォーマンスに悪影響を与えます。Repository に接続するユーザーがいない夜間などに、定期的にこれらのフォルダ内のファイルを消去することをお勧めします。
- **ダイアグラムとオブジェクトのチェックイン/チェックアウト**：RepoSrvDb.Log の情報を使用すると、パフォーマンスを最適化して、ユーザーが Repository をより効率的に使用できるようにアドバイスすることができます。たとえば、ダイアグラム全体をチェックイン/チェックアウトするよりも、複数のオブジェクトのみを選択してチェックイン/チェックアウトするほうが効率的です。RepoSrvDb.Log ファイルは、Repository のアプリケーションサーバー上に保存されています。Repository の使用ユーザー名と使用日時、発行された要求タイプ、処理結果、および要求の合計処理時間が記録されます。各要求の合計処理時間には、要求ファイルの解析、データベースからのデータの挿入/取得、および結果ファイルの構築と書き出しにかかった時間が含まれています。
- **Oracle 環境の場合**：
  - **複数の表領域を使用する**：Repository のテーブルとインデックスに対して、それぞれ ERSTUDIO\_DAT および ERSTUDIO\_IDX という個別の表領域を作成します。これらの表領域のエクステントは 1MB 以上に設定してください。大規模な Repository 環境では、さらにテーブルとインデックス自体を複数の表領域に分割すると良い場合もあるでしょう。たとえば、上で説明した表領域に加えて、ERSTUDIO\_DAT2 および ERSTUDIO\_IDX2 という表領域を追加します。
  - **インデックスを再構築する**：Repository のインデックス、ERSTUDIO\_IDX は、少なくとも月に一回は再構築することをお勧めします。追加されたダイアグラムの数によっては、週に一回のペースで再構築するのが望ましい場合もあります。
  - **統計情報を分析および計算する**：Repository に対するクエリーの応答時間を改善するには、Oracle の最近のバージョンでは DBMS\_STATS、古いバージョンでは Analyze Table および Compute Statistics 文を実行します。これらのコマンドは、テーブルとインデックス内のデータ分布についての情報を収集して、データ ディクショナリの情報を更新します。更新されたデータ ディクショナリを使用して、コストベースのクエリー オプティマイザは、テーブルやインデックスにアクセスする SQL 文の効率的な処理方法を決定できます。
  - **ログ設定を変更する**：チェックイン操作の実行時にチェックポイントが発生することがあり、それによって他の Repository アクションが中断して、応答時間が遅延することがあります。チェックポイントによる影響を最小限に留めるには、ログのサイズを増加させて、init.ora ファイル内の log\_checkpoint\_interval パラメータをゼロに設定し、ログバッファのサイズを小さくします。

## トラブル発生に備える

Repository の最適なパフォーマンスと可用性を確保するには、Repository スキーマを定期的にバックアップしておき、ハードウェア障害が発生した場合に、別のサーバーをバックアップサーバーに指定します。必要に応じて、新しい Repository サービスをバックアップサーバーにインストールして、その Repository を最新バックアップの復旧先に指定します。

## Repository でレポートを実行する

Repository ディレクトリ内の Utilities フォルダには、さまざまな SQL スクリプトが保存されています。SQL Agent を使用して、これらのスクリプトを実行できます。

## Repository をバックアップおよび復元する

Repository Database に対して、バックアップと復元の処理を実装しておくことをお勧めします。Repository の重要なデータは、Repository Server の Data フォルダに保存されます。

ローカルで作成して管理している \*.DM1 ファイルを Repository Server に追加すると、そのファイルは Repository Database (Oracle、Sybase、SQL Server、または DB2 のテーブル内に存在) で管理されます。これらのファイルは、ER/Studio Repository のインストールディレクトリで見つけることができます。デフォルトのインストール場所を指定した場合、ファイルは次の場所にあります。

- ...\\Embarcadero\\ERStudio Repository\\Data

**メモ：** ER/Studio DA ではオーバーフロー ファイルを使用しないようになりました。ロングファイルとして [Repository Database](#) に追加されます。

Repository Server は、Repository Database に対する専用のインターフェイスです。Repository に保存されたデータの状態を管理する役割があります。基本的に、Repository Server はデータベースに対するトランザクションサーバーとして動作します。Repository Database には、トランザクション履歴の他にも、セキュリティのルールとデータがすべて保存されます。Repository Server はデータベースとの間でデータのトランザクションを管理するため、各 Repository Database と Repository Server は 1 対 1 で対応する必要があります。

## Repository にクエリーを実行する

Repository にクエリーを実行して、SELECT 文による簡単なレポートを作成できます。次に示す例は、ダイアグラム、ダイアグラム内のサブモデル、および各サブモデル内のエンティティを選択する簡単なダイアグラム間レポートです。

**メモ：** データの編集、削除、または更新を行う場合は、Repository を使用しないでください。これらの操作を実行するには、ER/Studio DA を使用してください。

## 例

この例は、ER/Studio DA の Repository メタモデルを参照して作成しました。このデータ モデルは、次に示すアプリケーションのインストールディレクトリ内に保存されています。

```
... \Program Files\Embarcadero\ERStudio Data Architect
X.X\SystemModels\Repository600.dml

SELECT dbo.DiagramVer.Name as "Diagram Name",
dbo.SubModelVer.Name as "Submodel Name",
dbo.EntityVer.Name as "Entity Name",
dbo.EntityVer.TableName as "Table Name"
FROM dbo.Diagram, dbo.DiagramVer, dbo.Entity, dbo.EntityVer, dbo.Model,
dbo.SubModelVer, dbo.SubModel
WHERE (dbo.Entity.LatestVersionID = dbo.EntityVer.EntityVerID AND
dbo.Model.ModelID = dbo.Entity.ModelID AND
dbo.SubModel.LatestVersionID = dbo.SubModelVer.SubModelVerID AND
dbo.Model.ModelID = dbo.SubModel.ModelID AND
dbo.DiagramVer.DiagramID = dbo.Model.DiagramID AND
dbo.Diagram.LatestVersionID = dbo.DiagramVer.DiagramVerID)
```

## Repository のセキュリティを確保する

Repository コンテンツのセキュリティを確保するには、セキュリティ センターでユーザーを作成して、特定のロールと権限を割り当てます。

**メモ：** 予想外のロックアウトを避けるために、Admin ユーザーは削除できないようになっています。

たとえば、JimB というユーザーを作成して、DBA という名前のロールを割り当てます。このロールには複数の権限が付与されますが、いずれの権限も論理モデルを変更することはできません。Repository 管理者は、JimB を Repository 内の任意のモデルに関連付けることができます。JimB はダイアグラムの論理モデルを表示することはできますが、変更することはできません。

管理者は、Repository 項目へのユーザー アクセスを、プロジェクト、ダイアグラム、モデル、サブモデル、およびデータ ディクショナリのレベルで制限できます。Repository へのアクセスを制限するには、管理者は新規ユーザーを作成してから、特定の権限を付与するデフォルトロールを割り当てるか、またはそのユーザーに割り当てる新規ロールを作成します。その後、管理者はユーザーに一部の Repository 項目へのフルアクセスを付与したり、他の項目へのアクセスを制限または禁止することができます。

管理者はユーザーとロールを作成する前に、次のような Repository の概念を理解しておく必要があります。

- **カスケード対応したセキュリティ**：Repository に多数のダイアグラムがある場合に、複数のユーザーに一括してロールを割り当てることができるように（たとえば、ユーザーが複数のダイアグラムに対して同じアクセス権限を持つ必要がある場合など）、管理者は ER/Studio Repository のさまざまな「レベル」にユーザーとロールを割り当てることができます。

たとえば、ER/Studio Repository で複数のダイアグラムを含むプロジェクトを作成するとします。ユーザーにこのプロジェクトに対するロールを割り当てると、プロジェクト内に含まれるすべてのダイアグラムに同じ権限セットが付与されます。ユーザーに「Repository」レベルでロールを割り当てると、それ以下のレベルにも権限がカスケードして付与されます。

同一ユーザーに低いレベルの異なるロールを割り当てることで、高いレベルのロールを上書きすることができます。たとえば、UserA は Repository レベルでは Viewer ロール（読み取り専用）を持ちますが、特定のダイアグラムやサブモデルに対しては Modeler ロール（編集可）を持つことができます。

- **クライアント側でのセキュリティ キャッシュ**：独立した状態でのモデリング共同開発を実現するために（つまり、ER/Studio Repository からログアウトされ、ネットワークからも切り離された状態で）、ユーザーが ER/Studio Repository にログインすると、ダイアグラムごとにユーザーに関連付けられているすべてのセキュリティが、クライアント上のキャッシュに保存されます。

**メモ**：セキュリティ設定が変更された場合、ユーザーは Repository に再ログインして権限を更新する必要があります。

- **スーパー ユーザー**：スーパー ユーザーは、ER/Studio Repository のインストール時に作成されるデフォルトのロールです。デフォルトの「Admin」ユーザーには、スーパー ユーザーが割り当てられています。スーパー ユーザーにはすべての権限が付与されているため、ER/Studio Repository で管理されているダイアグラム、ユーザー、ロールのすべてにアクセスできます。スーパー ユーザーを削除したり変更することはできません。また、セキュリティセンターの [Repository のセキュリティ] タブにある「Repository」レベルにのみ表示されます。
- **アクセス権なし**：「アクセス権なし」は、Repository のプロジェクト レベルとダイアグラム レベルでのみ有効なロールです。このロールを使用すると、特定のプロジェクト内で管理されている複数のダイアグラムや、個別のダイアグラムに対して、いかなるアクセスも許可しないように簡単に設定できます。

## メモ

- ユーザーが Repository からファイルをローカルのアクティブ ファイルディレクトリに取得すると、取得されたファイルにユーザーとマシンの組み合わせが記録されます。さらに、ユーザーが「オフライン」状態や Repository Server に接続できない場合でもモデルを作成できるようにするため、ER/Studio DA では、ユーザーのセキュリティ権限とダイアグラムに関する情報がアクティブ ファイルディレクトリに取得されたファイルに保存されます。これによって、次のような状況が発生します。
- あるユーザーとマシンの組み合わせで Repository から取得されて、アクティブ ファイルディレクトリに保存されている DM1 ファイルを、別ユーザーでログインしてから開こうとすると、競合が発生してファイルを開くことができないというメッセージが表示されます。これは、別のユーザーが先に取得しているダイアグラムを開いて作業したり、変更をチェックインできないようにするためです。Repository では、オブジェクトのチェックアウト時にユーザーとマシンの組み合わせが記録されるため、オブジェクトを最初にチェックアウトしたマシンのユーザーのみが、そのオブジェクトをチェックインできます。
- ユーザーは、Repository にログインしていない状態でも、Repository DM1 ファイルを ER/Studio DA でファイルを開いて作業できます。ER/Studio DA では、最初に Repository からファイルを取得したユーザーのセキュリティ権限データを保存しています。このセキュリティデータにより、ユーザーが Repository にログインするかファイルを開じるまで、残りの作業セッションが管理されます。ユーザーが Repository にログインしていないときに Repository から取得された複数のファイルを開く場合、これらのファイルはすべて同一ユーザーにより取得されたものである必要があります。別ユーザーが取得したファイルを開くと、競合が発生してファイルを開くことができないというメッセージが表示されます。もちろん、ユーザーがログインしている場合も別ユーザーがローカルに保存したファイルを開くことはできません。
- 各ダイアグラムと共にキャッシュに保存されたセキュリティ データが更新されるのは、Repository へのログイン中に、開いている Repository ファイルが保存された時点です。そのため、ファイルを開じる前に保存していない場合は、そのユーザーとダイアグラムに対してセキュリティ権限が変更されていても、ファイルと共にキャッシュに保存されることはありません。オフラインで作業する時に、ファイルが最後に保存された後にセキュリティ設定が変更された場合は、セキュリティ情報がローカルに保存したファイルに保存されていません。Repository にログインして作業対象の各ファイルを開き保存してからオフラインで作業する必要があります。
- 権限設定をプロジェクト レベルまたは Repository レベルで適用している場合、権限設定を変更するタイミングによってオフライン作業時に 2 つのファイル間で競合が発生することがあります。2 つのローカル ファイルが開かれたときに、それぞれのファイルに Repository レベルまたは同じプロジェクトに対する異なるセキュリティ情報がキャッシュされている場合は、最も新しく保存されたデータが使用され、ファイルの保存時に両方のファイルに格納されます。
- 管理者がセキュリティ センターで変更を加えた場合、たとえばフォルダの権限を変更してアクセス権を付与または取り消したり、ダイアグラムを異なる権限が設定された別のプロジェクトに移動すると、ユーザーは Repository に再ログインしてローカルのモデルに保存された権限の情報を更新する必要があります。

このセクションには次のトピックが含まれています。

- [ルールを作成および管理する](#) (30 ページ)
- [ユーザーを作成および管理する](#) (36 ページ)
- [Repository 項目へのユーザー アクセスを付与および禁止する](#) (38 ページ)

## ルールを作成および管理する

ER/Studio DA では、さまざまな権限セットを割り当てたカスタマイズルールを作成することができます。[Repository のオブジェクトタイプの権限] 領域 (Repository セキュリティセンターの [ルールの管理] タブ) には、Repository 項目を操作するための事前定義された権限の一覧が含まれています。この一覧から必要な権限をルールに割り当てた後に、そのルールをユーザーに割り当てることで、特定の Repository 操作を実行する権限がユーザーに付与されます。さまざまな権限を組み合わせ、ご使用の環境に適したルールを作成できます。

たとえば、次のようなルールを作成して割り当てることができます。

- **基本 Repository レベル** : Repository レベルのすべての権限があります。[Repository のセキュリティ] タブの [Repository オブジェクト] 領域で Repository レベルを選択して、対象ユーザーに割り当てます。
- **論理モデリングのみ** : ダイアグラム、モデル、およびサブモデル レベルのすべての権限があります。ただし、これらの権限は論理モデルに限定されます。[Repository のセキュリティ] タブの [Repository オブジェクト] 領域で、Repository 内の各ダイアグラムのレベル、または作業を許可するダイアグラムのレベルを選択して、対象ユーザーに割り当てます。

上の2つのルールが割り当てられたユーザーがダイアグラムを追加すると、そのユーザーはダイアグラムを取得できますが、変更することはできません。ダイアグラムを追加した直後に、ユーザーが論理モデルを変更できるようにするには、「基本 Repository レベル」ルールに「論理モデリングのみ」ルールの内容を追加してからユーザーに適用し、Repository レベルで論理モデルを変更できるようにする必要があります。

**メモ** : Repository 管理者がユーザー情報にアクセスして、作成、更新、および削除操作を実行するには、[Repository のオブジェクトタイプの権限] 内の [セキュリティ情報へのアクセス] および [セキュリティ情報の更新] の権限が必要です。

### ルールを作成、更新、および削除する

1. [Repository] メニューの [セキュリティ] をポイントし、[セキュリティセンター] をクリックします。
2. [ルールの管理] タブをクリックします。
3. [新規作成] をクリックします。

4. [Repository のオブジェクトタイプ] 一覧の項目をクリックして、次に [Repository のオブジェクトタイプの権限] 一覧で、選択したオブジェクトタイプに関する権限をロールに割り当てます。
5. [Repository のオブジェクトタイプ] 一覧の各項目に対して、[手順4](#) を繰り返します。
6. [適用] をクリックして、引き続きセキュリティ設定を変更してから、[OK] をクリックしてセキュリティセンターを閉じます。
7. 既存ユーザーに新しいロールを割り当てるには、36 ページの「[ユーザーにロールを割り当てる](#)」を参照してください。
8. 設定の変更によって影響を受けるユーザーに、Repository から一度ログアウトして再ログインするよう通知します。ユーザーが保持しているセキュリティ情報を更新するためです。

## メモ

- 作成したロールを選択して、更新、削除、または名前の変更をいつでも実行できます。ロールの名前を変更しても、ユーザーの権限には影響しませんが、ロールを削除すると、そのロールを通じて Repository 項目へのアクセス権を持つユーザーは、それらの項目にアクセスできなくなります。
- ユーザーがオブジェクトをチェックアウトしているかどうかに関係なく、必要な権限がない場合はオブジェクトを変更できません。
- 新しく追加されたダイアグラム（プロジェクトではなく）にすぐにアクセスできるようにするには、管理者は次のような権限を設定する必要があります。ユーザーがサブモデルを取得してエンティティを追加する場合、そのユーザーには、そのサブモデルを含むモデルの [ダイアグラム オブジェクトの作成] 権限、およびそのサブモデルの [メンバの追加] 権限が必要です。ユーザーがエンティティを削除する場合、そのユーザーには、そのサブモデルの [メンバの削除] 権限が必要です。さらに、ユーザーがサブモデルからエンティティを削除するときに [モデルから削除] チェック ボックスをオンにする場合、そのユーザーには、そのモデルの [ダイアグラム オブジェクトの削除] 権限が必要です。
- Repository からロールを削除すると、そのロールはデータベースには存在しなくなります。
- ユーザー情報にアクセスして、作成、更新、および削除操作を実行するには、Repository 管理者のロールに「セキュリティ情報へのアクセス」と「セキュリティ情報の更新」権限を割り当てる必要があります。
- ロールを削除するときは、あらかじめロールに割り当てられたダイアグラムのリンクを解除して、ロールに割り当てられたユーザーを除去する必要があります。

- 次の表に、Repository のオブジェクトタイプ、ユーザーに付与できる権限、およびこれらの権限でアクセスできる一般的な操作を示します。

Repository のオブジェクトタイプ	Repository のオブジェクトタイプの権限	許可される操作
Repository	セキュリティ情報へのアクセス	セキュリティセンター：設定の確認
	セキュリティ情報の更新	セキュリティセンター：設定の確認と変更
	ダイアグラムの作成	ダイアグラムの追加
	ダイアグラムの更新	ダイアグラムのチェックアウト、ダイアグラムのチェックイン オブジェクトのチェックアウト、オブジェクトのチェックイン ダイアグラムのチェックアウトの取り消し、オブジェクトのチェックアウトの取り消し ダイアグラムのチェックアウトのやり直し、オブジェクトのチェックアウトのやり直し
	ダイアグラムの削除	ダイアグラムの削除 (Repository 内のダイアグラムを削除した場合でも、ファイル自体はローカルディレクトリから削除されません。)
	エンタープライズディクショナリの作成	エンタープライズディクショナリの作成
	エンタープライズディクショナリの更新	データディクショナリのチェックアウト、ディクショナリオブジェクトのチェックアウト、データディクショナリのチェックイン、データディクショナリのチェックアウトの取り消し、データディクショナリのチェックアウトのやり直し
	プロジェクトの作成	プロジェクトの作成
	プロジェクトの削除	プロジェクトの削除
	プロジェクト	プロジェクトメンバの追加
プロジェクトメンバの削除		ダイアグラムをプロジェクトから削除



Repository のオブジェクトタイプ	Repository のオブジェクトタイプの権限	許可される操作
ダイアグラム	エンタープライズディクショナリのバインド	新規エンタープライズデータディクショナリの作成 既存のエンタープライズデータディクショナリのバインド
	エンタープライズディクショナリのアンバインド	エンタープライズデータディクショナリの除去
	モデルの比較	比較とマージウィザードの実行
	物理モデルの作成	物理モデルの作成
	物理モデルの削除	モデルの削除
	名前付きリリースの設定	名前付きリリースの設定
	名前付きリリースの削除	名前付きリリースの削除
	名前付きリリースへのダイアグラムのロールバック	ダイアグラムのロールバック
	ダイアグラムプロパティの更新	タイトルブロックデータの編集、ダイアグラムプロパティの編集
	データフローの作成	[データリネージ] タブに新規データフローを作成
	データフローの削除	[データリネージ] タブからデータフローを削除
データディクショナリ	ディクショナリオブジェクトの作成	<b>作成:</b> アタッチメントタイプ、アタッチメント、デフォルト、ルール、データ移動ルール、参照値、ユーザーデータ型、ドメインフォルダ、ドメイン、再利用可能なトリガー、再利用可能なプロシージャ、ライブラリ
	ディクショナリオブジェクトの更新	<b>編集:</b> アタッチメントタイプ、アタッチメント、デフォルト、ルール、データ移動ルール、参照値、ユーザーデータ型、ドメインフォルダ、ドメイン、再利用可能なトリガー、再利用可能なプロシージャ、ライブラリ
	ディクショナリオブジェクトの削除	<b>削除:</b> アタッチメントタイプ、アタッチメント、デフォルト、ルール、データ移動ルール、参照値、ユーザーデータ型、ドメインフォルダ、ドメイン、再利用可能なトリガー、再利用可能なプロシージャ、ライブラリ

Repository のオブジェクトタイプ	Repository のオブジェクトタイプの権限	許可される操作
論理メインモデル	ダイアグラム オブジェクトの作成	<b>作成</b> : エンティティ、ビュー、リレーションシップ、ビューリレーションシップ、サブタイプクラスタ、サブタイプ、タイトルブロック
	ダイアグラム オブジェクトの削除	<b>削除</b> : モデルのエンティティ、ビュー、リレーションシップ、ビューリレーションシップ、サブタイプクラスタ、サブタイプ、タイトルブロック
	ダイアグラム オブジェクトの更新	<b>エンティティエディタ</b> : 属性、キー、キー属性、および CHECK 制約の作成 / 変更 / 削除 <b>ビューエディタ</b> : ビューの変更、ビューテーブルおよびビューカラムの作成 / 変更 / 削除 <b>キーエディタ</b> : キーの変更、キー属性の作成 / 変更 / 削除 <b>リレーションシップエディタ</b> : リレーションシップの変更 <b>サブタイプクラスタエディタ</b> : サブタイプクラスタの変更 <b>モデルの編集</b> : オプション、プロパティ
	サブモデルの作成	サブモデルの作成
	サブモデルの削除	サブモデルの削除
論理サブモデル	メンバの追加	<b>サブモデルエディタ</b> : サブモデルに追加
	メンバの削除	<b>サブモデルエディタ</b> : サブモデルから除去、サブモデルからエンティティを削除 データベースビューの除去 <b>削除</b> : サブモデルのリレーションシップ、ビューリレーションシップ、サブタイプクラスタ、およびサブタイプ
	表示プロパティの更新	<b>移動/サイズの最適化</b> : エンティティ/テーブル、ビュー、タイトルブロック、テキストブロック、サブタイプクラスタ <b>色/フォントの変更</b> : エンティティ/テーブル、ビュー、タイトルブロック、テキストブロック、リレーションシップライン、ビューリレーションシップライン、サブタイプクラスタ <b>移動</b> : リレーションシップライン、ビューリレーションシップライン <b>作成/変更/削除</b> : テキストブロック モデル表記法の変更、レイアウト、ズーム、およびオブジェクト整列の実行
物理モデル	ダイアグラム オブジェクトの作成	<b>作成</b> : テーブル、ビュー、リレーションシップ、ビューリレーションシップ、スキーマオブジェクト、タイトルブロック
	ダイアグラム オブジェクトの削除	<b>削除</b> : テーブル、ビュー、リレーションシップ、ビューリレーションシップ、スキーマオブジェクト、タイトルブロック
	ダイアグラム オブジェクトの更新	<b>作成/変更/削除</b> : カラム、インデックス、インデックスカラム、CHECK 制約、ビューテーブル、ビューカラム、キー属性、キー、リレーションシップ、サブタイプクラスタ <b>モデルの編集</b> : オプション、プロパティ 対象データベースの変更
	サブモデルの作成	サブモデルの作成
	サブモデルの削除	サブモデルの削除

Repository のオブジェクトタイプ	Repository のオブジェクトタイプの権限	許可される操作
物理サブモデル	メンバの追加	<b>サブモデルエディタ</b> ：サブモデルに追加
	メンバの削除	<b>サブモデルエディタ</b> ：サブモデルから除去 <b>削除</b> ：サブモデルのテーブル、ビュー、リレーションシップ、ビューリレーションシップ、スキーマオブジェクト
	表示プロパティの更新	<b>移動/サイズの最適化</b> ：エンティティ/テーブル、ビュー、タイトルブロック、テキストブロック、物理スキーマオブジェクト <b>色/フォントの変更</b> ：エンティティ/テーブル、ビュー、タイトルブロック、テキストブロック、リレーションシップライン、物理スキーマオブジェクト <b>移動</b> ：リレーションシップライン、ビューリレーションシップライン <b>作成/変更/削除</b> ：テキストブロック モデル表記法の変更、レイアウト、ズーム、およびオブジェクト整列の実行
データフローモデル	データフローオブジェクトの作成	<b>作成</b> ：データフロー、データリネージコンポーネント、データストリーム、ソース
	データフローオブジェクトの更新	<b>編集</b> ：データフロー、オブジェクト、変換、データストリーム、ソース <b>チェックイン</b> ：データフロー、オブジェクト、変換、データストリーム、ソース <b>チェックアウト</b> ：データフロー、オブジェクト、変換、データストリーム、ソース <b>チェックアウトの取り消し</b> ：データフロー、オブジェクト、変換、データストリーム、ソース <b>チェックアウトのやり直し</b> ：データフロー、オブジェクト、変換、データストリーム、ソース <b>メモ</b> ：ダイアグラム内のデータフローオブジェクトを更新するには、[ダイアグラムの更新] 権限が必要です。
	データフローオブジェクトの削除	<b>削除</b> ：データフロー、データリネージコンポーネント、変換、データストリーム <b>メモ</b> ：データリネージウィンドウからテーブル/エンティティコンポーネントを削除しても、モデルのテーブル/エンティティは削除されません。 <b>メモ</b> ：ダイアグラム内のデータフローオブジェクトを削除するには、[ダイアグラムの削除] 権限が必要です。
データフローの表示	表示プロパティの更新	<b>移動/サイズの最適化</b> ：変換、コンポーネント、データフロー <b>色/フォントの変更</b> ：データリネージの背景色、コンポーネント、変換、データストリーム <b>ダイアグラムとオブジェクトの表示オプション</b> ：このダイアログボックスのオプションを変更して、データリネージダイアグラムの表示形式を設定できます。 レイアウト、ズーム、オブジェクト整列、データストリームのレイアウト、水平/垂直方向に直線化、すべての頂点を除去

## ユーザーにロールを割り当てる

ユーザーにロールを割り当てると、ロールに指定されたすべての権限がユーザーに付与されます。

1. [Repository] メニューの [セキュリティ] をポイントし、[セキュリティ センター] をクリックします。
2. ユーザーにロールが割り当てられていない場合、[Repository のセキュリティ] タブで、[利用可能なユーザー] 領域のユーザー名を [利用可能なロール] 領域のロール名にドラッグアンドドロップします。

ユーザーにロールを割り当て済みの場合、[Repository のセキュリティ] タブで、[利用可能なロール] 領域のユーザー名を、別のロール名にドラッグアンドドロップします。

3. [適用] をクリックしてから、[OK] をクリックしてセキュリティ センターを閉じます。
4. 一度ログアウトしてから再ログインするようにユーザーに通知します。ユーザーが保持しているセキュリティ情報を更新するためです。

## ユーザーを作成および管理する

Repository のユーザー管理機能を使用すると、ユーザーのログイン情報や権限を作成、変更、および削除することができます。Repository 環境で問題が発生した際には、ユーザーがチェックアウト中のオブジェクトをチェックインできます。また、ユーザーをロールに関連付けることで、特定のダイアグラム、モデル、サブモデル、またはエンタープライズ データ ディクショナリに対する権限を管理できます。ロールを使用すると、Repository 項目に対するユーザーの権限を自由に設定できます。

**メモ:** Repository 管理者がユーザー情報にアクセスして、作成、更新、および削除操作を実行するには、[Repository のオブジェクト タイプの権限] 内の [セキュリティ情報へのアクセス] および [セキュリティ情報の更新] の権限が必要です。

### ユーザーの管理操作を実行する（作成、編集、削除、非アクティブ化、再アクティブ化、ログアウト、およびチェックイン）

1. [Repository] メニューの [セキュリティ] をポイントし、[セキュリティ センター] をクリックします。
2. [ユーザーの管理] タブをクリックします。
3. 一覧からユーザー名を選択して、[ユーザー管理] および [管理者ユーザーによる強制オプション] 領域のボタンをクリックし、必要な管理操作を実行します。
4. 必要に応じて、[Repository のセキュリティ] タブでセキュリティ設定を変更してから、[OK] をクリックしてセキュリティ センターを閉じます。
5. ユーザーに権限を割り当てるには、[「ユーザーにロールを割り当てる」](#) を参照してください。

6. セキュリティ設定を変更した場合は、一度ログアウトしてから再ログインするようにユーザーに通知します。ユーザーが保持しているセキュリティ情報を更新するためです。

次に、一部のオプションについて補足説明します。

### [ユーザー管理] 領域

- [新規作成] : [Repository ユーザーの作成] ダイアログ ボックスが開きます。
- [ディレクトリ サービス ユーザー] : ユーザーが Windows 認証を使用して Repository にログインする場合は、このチェック ボックスをオンにします。ER/Studio DA はネットワーク上の LDAP サーバーにアクセスして、ユーザーの資格情報を検証します。ディレクトリ サービス ユーザーは、[Repository へのログイン] ダイアログ ボックスで [現在の Windows アカウントでログインする] チェック ボックスをオンにして、Windows のユーザー ID とパスワードを使用して Repository にログインできます。
- [Repository ユーザーの強制チェックイン] : [ユーザーのチェックイン] をクリックすると、チェックアウト中のオブジェクトの処理を選択できます。
- [非排他的に切り替える] : ユーザーは引き続きチェックアウト中の項目に変更を加えることができます。他のユーザーも同じ項目をチェックアウトできるようになります。非排他的にチェックアウトした項目をチェックインするときに、Repository 内のその項目が他のユーザーによって変更済みの場合は、ユーザー自身の変更を上書き保存するか、または他のユーザーの変更をそのまま使用するかを選択して、相違点を解決できます。
- [チェックアウトの取り消し] : ユーザーが次に Repository に接続したときに、そのユーザーがチェックアウト中の項目が、チェックアウトされていない状態に更新されます。

### メモ

- Admin ユーザーは削除したり非アクティブ化することはできません。Admin ユーザーのデフォルトのパスワード (「Admin」) は変更しておくことをお勧めします。
- ユーザーの一覧に表示された新規ユーザーの名前には、星印のアイコンが付きます。これは、Repository 項目に対するロールがまだ割り当てられていないことを示しています。
- 更新されたセキュリティ設定をユーザーが取得するには、一度ログアウトしてから再ログインする必要があります。
- Admin は、ユーザーの現在のパスワードを知らなくても、そのユーザーの新しいパスワードを設定できます。
- Admin は [ユーザーの管理] タブを確認して、Repository にログイン中のユーザーと、そのユーザーが Repository 項目をチェックアウトしているかを知ることができます。ユーザーのチェックアウト情報は、Repository をアップグレードする際にも役立ちます。Repository をアップグレードする前には、すべての項目をチェックインする必要があります。ログインユーザーの情報は、セキュリティ センターの設定を変更する際に役立ちます。設定の変更後、Admin はオンライン ユーザーに対して一度ログアウトしてから再ログインするよう通知して、更新されたセキュリティ情報を取得してもらうことができます。

- 管理者は、セキュリティセンターの設定更新を取得してもらうために、ユーザーを強制的にログアウトさせることができます。たとえば、プロジェクトに新しいダイアグラムを追加して、プロジェクトへのアクセス権限を持つすべてのユーザーがそのダイアグラムにアクセスできないように設定する必要があるとします。管理者はプロジェクトのセキュリティ設定を変更した後に、ユーザーを強制的にログアウトさせて、再ログインしてもらうことでユーザーのセキュリティ情報を更新することができます。その他の場合でも、管理者は必要に応じてユーザーを Repository からログアウトさせることができます。

## Repository 項目へのユーザー アクセスを付与および禁止する

ER/Studio Repository セキュリティセンターの [Repository のセキュリティ] タブで、適切な権限を持つユーザーは他のユーザーにさまざまなオブジェクトへのアクセス権を付与できます。Repository オブジェクトに特定の操作を実行する権限を与えたり、そのオブジェクトに他の操作を実行することを禁止することができます。

### Repository オブジェクトへのアクセス ロールをユーザーに割り当てる

1. Repository にログインします。
2. [Repository] メニューの [セキュリティ] をポイントし、[セキュリティセンター] をクリックします。
3. [ロールの管理] タブをクリックして、利用可能なロールを参照し、ユーザーに付与したいアクセスセットが設定された既存のロールがあるかを確認します。必要に応じて、新しいロールを作成して適切な権限を設定します。
4. [Repository のセキュリティ] タブをクリックします。
5. [Repository オブジェクト] 領域で、アクセスを付与または禁止する Repository 項目（ダイアグラム、オブジェクト、サブモデル、またはデータディクショナリ）を選択します。
6. [利用可能なユーザー] 領域で、選択したオブジェクトへのロールを割り当てるユーザーを選択して、[利用可能なロール] 領域内の適切なロールにドラッグします。

**ヒント:** Repository 項目に対するユーザーのロール割り当てを削除するには、Repository 項目をクリックして、[利用可能なロール] 領域で、ロールの下に配置されたユーザー名を [利用可能なユーザー] 領域にドラッグします。

### メモ

- 1人のユーザーに割り当てることができるロールは、各オブジェクトに対して1つだけです。ただし、異なるオブジェクトに対しては、異なるロールを割り当てることができます。
- [利用可能なユーザー] 領域のユーザーが灰色表示になっている場合、そのユーザーは非アクティブ化されています。編集するにはユーザーを再アクティブ化する必要があります。

### 特定の Repository 項目へのユーザー アクセスを禁止する

システム定義の「アクセス権なし」ロールを使用すると、指定したユーザーが特定のプロジェクトやダイアグラムへアクセスするのを禁止できます。

1. Repository にログインします。
2. [Repository] メニューの [セキュリティ] をポイントし、[セキュリティ センター] をクリックします。
3. [Repository オブジェクト] 領域で、指定したユーザーからのアクセスを禁止したいプロジェクトまたはダイアグラムを選択します。
4. [利用可能なユーザー] 領域の一覧から、ユーザー名をクリックするか、または <Ctrl> キーを押しながら複数のユーザー名をクリックして、[利用可能なロール] 領域の [アクセス権なし] ロールにドラッグします。
5. 他のダイアグラムやプロジェクトへのアクセスも禁止する場合は、[手順 3](#) と [手順 4](#) を繰り返します。
6. [適用] をクリックして、必要に応じてセキュリティ設定をさらに変更するか、または [OK] をクリックしてセキュリティセンターを閉じます。

### Repository 項目へのユーザー アクセスを変更する

Repository 項目へのユーザー アクセスを変更するには、その Repository 項目に対するユーザーロールの権限を変更するか、またはその Repository 項目用に設定した別のロールをユーザーに割り当てます。必要に応じて、ユーザーを強制的にログアウトさせてから、ユーザーのアクセス権を変更することもできます。

