

PROVA DE CONCEITO - FERRAMENTAS DE IMPLANTAÇÃO DE BANCO DE DADOS

1. INFORMAÇÕES GERAIS DO PROJETO

Responsável pela Avaliação: Thiago Fernandes da Silva Oliveira

Período de Avaliação: 18/02/2026 – 18/03/2026

Versão do Documento: 1.0

Data de Elaboração: 18/02/2026

2. OBJETIVO DA PROVA DE CONCEITO

- **Objetivo principal:** documentar comparações de ferramentas para auxiliar no gerenciamento de versão de banco de dados através de documentações e resultados de prova de conceito.
- **Escopo da avaliação:** Comparar informações existentes de mercado e comunidade, realizar as provas de conceito: Criação de Estrutura Inicial, Alteração de Estrutura Existente, Desfazer Mudanças, Integração com Pipeline CI/CD).
- **Ferramentas avaliadas:** Flyway, Liquibase, SQL Server Data Tools.
- **Ambientes de teste utilizados:** Agentes Jenkins (Máquina Virtual e *Container*) e Estação de Trabalho.

3. CONTEXTO E NECESSIDADES

- **Descrição do ambiente atual:** implantação atual é realizada através do *SQL Server Management Studio* com scripts armazenados no *Subversion* (SVN).
- **Problemas identificados:** falta de versionamento do banco de dados, falta de processo de aprovação de scripts no repositório de versionamento, falta de automatização na implantação.
- **Requisitos técnicos obrigatórios:** solução permitir versionamento do banco de dados em repositórios git, permitir execução através de pipelines jenkins, permitir consultar histórico de implantações.
- **Requisitos desejáveis:** permitir observabilidade, logs detalhados.
- **Restrições e limitações conhecidas:** teste possível apenas em ambiente de desenvolvimento, não impactar em bases além da base de teste.

4. FERRAMENTAS AVALIADAS

Flyway

- **Versão testada:** Flyway Enterprise Edition 12.0.2-rc1522
- **Modo de instalação:** Arquivo .tar.gz
- **Licenciamento:** Gratuito, Teams e Enterprise
- **Descrição resumida:** é uma ferramenta de migração e versionamento de banco de dados open-source, para gerenciar alterações de esquema (tabelas, colunas) de forma automatizada, segura e controlada. Ele atua como um "Git para banco de dados", executando scripts SQL versionados (V1__desc.sql, U1__desc.sql) sequencialmente, ideal para ambientes colaborativos e CI/CD.
- **Pontos fortes observados:** Mantém tabela com histórico de mudanças
- **Pontos fracos observados:**
- **Documentação e suporte:** <https://documentation.red-gate.com/fd/quickstart-flyway-cli-252281654.html>

Liquibase

- **Versão testada:** 5.0.1 #9400
- **Modo de instalação:** Gerenciadores de pacote Linux apt e yum.
- **Licenciamento:** Community, Pro e Secure
- **Descrição resumida:** é uma ferramenta de código aberto para gerenciamento e versionamento de migrações de esquema de banco de dados (*database schema*

migrations). Ele permite rastrear, controlar e implantar alterações no banco de dados de forma automática e segura, suportando SQL, XML, YAML e JSON. Facilita o desenvolvimento e a integração contínua (CI/CD) em diferentes ambientes, evitando conflitos.

- **Pontos fortes observados:**
- **Pontos fracos observados:**
- **Documentação e suporte:** <https://docs.liquibase.com/community/user-guide-5-0>

SQL Server Data Tools (SSDT)

- **Versão testada:** Visual Studio 2022, SSDT 17.0.62207.28050 e SQL Package 170.2.70.1.
- **Modo de instalação:** Visual Studio Instalador e dotnet tool para SQL Package.
- **Licenciamento:** Gratuito, mas precisa do SQL Server Licenciado.
- **Descrição resumida:** Pacote que permite criar projeto de banco de dados.
- **Pontos fortes observados:** permite focar em criar apenas a estrutura dos objetos de banco de dados, permite implantar mudanças sem ter que gerar os scripts de alteração, possui proteção com deleções de objeto (precisa ser liberado explicitamente via comando ou configuração).
- **Pontos fracos observados:** o projeto precisa de todos objetos de banco de dados existentes, em sistemas que criam tabelas dinamicamente pode impactar na exclusão das tabelas.
- **Documentação e suporte:** <https://learn.microsoft.com/pt-br/sql/ssdt/sql-server-data-tools?view=sql-server-ver17>

5. CRITÉRIOS DE AVALIAÇÃO

Critério	Flyway	Liquibase	SSDT	Observações
Facilidade de instalação e configuração	Necessário ajuste de Dockerfile inserindo os devidos comandos	Necessário ajuste de Dockerfile inserindo os devidos comandos	Instalador passo a passo	
Curva de aprendizado	Mantém o formato de scripts de implantação atual	Permite uso de SQL juntando script de alteração junto com rollback	Ideal para equipes que já trabalham com Visual Studio e SQL Server	
Integração com CI/CD	Possui plugin Necessário criar agente	Necessário criar agente	Possível através de agente jenkins com SSDT e SQLPackage	
Suporte a rollback	Possui nomenclatura específica para scripts de rollback, execução manual	Automático para alguns casos	Manual	
Versionamento de scripts	Versionar scripts para alterar banco existente	Versionar scripts para alterar banco existente	Versionar os scripts do banco inteiro	SSDT mantém projeto do banco inteiro, não precisa de scripts
Gestão de mudanças (changesets)	Guarda registro de versão atual e	Guarda registro de versão atual e	Possível através de ferramenta de versionamento	

	histórico de changesets	histórico de changesets		
Performance na execução	Alta	Alta	Média	O SSDT gera dinamicamente as mudanças, por isso leva um tempo maior durante implantação.
Suporte a múltiplos bancos de dados	SQL	SQL e alguns NoSQL	Somente SQL Server	
Rastreabilidade e auditoria	Log de execução do jenkins Tabela de histórico mudanças	Log de execução do jenkins Tabela de histórico de mudanças	Log exibe todos objetos que foram alterados	
Gestão de ambientes (Dev, QA, Prod)	Histórico de mudanças salvo em cada ambiente	Histórico de mudanças salvo em cada ambiente	Deve se atrelar de forma externa versões do banco de dados em cada ambiente	
Documentação disponível	Documentação oficial do Redgate	Documentação oficial do Liquibase e cursos oficiais	Documentação Oficial da Microsoft	
Comunidade e suporte	Suporte Oficial e comunidade github	Suporte Oficial e comunidade github	Suporte oficial da Microsoft e forum de discussão	
Custo de licenciamento	Community - Sem limites de schemas, não suporta rollback Teams - \$3k/ano, 100 schemas produção Enterprise - ~\$10k/ano, sem limite de schemas	Pro - ~ \$5k/ano, mínimo 10 alvos Enterprise – sob consulta, alvos ilimitados	Gratuito	
Integração com ferramentas existentes	Git CI/CD Kubernetes Prometheus / Grafana Redgate docs para métricas CI	Git CI/CD Kubernetes Prometheus / Grafana Log estruturado Relatórios de drift	Git CI/CD Logs SQLPackage	
Facilidade de manutenção	Atualização de agente jenkins	Atualização de agente jenkins	Atualização no agente jenkins	

6. CENÁRIOS DE TESTE EXECUTADOS

Foram definidos 3 cenários de teste envolvendo criação, alteração de objetos e pipeline CI/CD, todos cenários envolvem o processo de implantação e rollback de banco de dados.

Todos os cenários consideram uma mesma base, mesmos objetos criados/atualizados, mesmo usuário de banco de dados.

Os arquivos utilizados nos cenários se encontra no repositório

https://gitlab.banpara.com/tfoliveira/octopussrvconexao/-/tree/principal?ref_type=heads.

■ Cenário 1: Criação Inicial de Estrutura

- **Descrição do teste:** Criação de 5 tabelas e rollback excluir as 5 tabelas
- **Artefatos criados:** tbCliente, tbContato, tbProduto, tbOportunidade, tbAtividade

```
CREATE TABLE [dbo].[tbCliente] (  
    [ClienteId] INT IDENTITY (1, 1) NOT NULL,  
    [Nome] VARCHAR (100) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,  
    [Email] VARCHAR (100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [Telefone] VARCHAR (20) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [DataCadastro] DATETIME2 (3) DEFAULT (sysutcdatetime()) NOT NULL,  
    [Status] VARCHAR (20) COLLATE SQL_Latin1_General_CP1_CI_AS DEFAULT  
    ('Potencial') NOT NULL,  
    [CpfCnpj] VARCHAR (20) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [Endereco] NVARCHAR (500) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    CONSTRAINT [PK_tbCliente] PRIMARY KEY CLUSTERED ([ClienteId] ASC),  
    CONSTRAINT [UK_tbCliente_Email] UNIQUE NONCLUSTERED ([Email] ASC),  
    CONSTRAINT [UK_tbCliente_CpfCnpj] UNIQUE NONCLUSTERED ([CpfCnpj] ASC),  
    CONSTRAINT [CK_tbCliente_Status] CHECK ([Status] IN  
    ('Ativo','Inativo','Potencial'))  
);  
  
CREATE TABLE [dbo].[tbContato] (  
    [ContatoId] INT IDENTITY (1, 1) NOT NULL,  
    [ClienteId] INT NOT NULL,  
    [Nome] VARCHAR (100) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,  
    [Cargo] VARCHAR (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [Email] VARCHAR (100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [Telefone] VARCHAR (20) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [Principal] BIT DEFAULT (0) NOT NULL,  
    CONSTRAINT [PK_tbContato] PRIMARY KEY CLUSTERED ([ContatoId] ASC),  
    CONSTRAINT [FK_tbContato_tbCliente] FOREIGN KEY ([ClienteId]) REFERENCES  
    [dbo].[tbCliente] ([ClienteId]) ON DELETE NO ACTION  
);  
  
CREATE TABLE [dbo].[tbProduto] (  
    [ProdutoId] INT IDENTITY (1, 1) NOT NULL,  
    [Nome] VARCHAR (100) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,  
    [Descricao] NVARCHAR (MAX) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [Preco] DECIMAL (10, 2) NOT NULL,  
    [Categoria] VARCHAR (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [Ativo] BIT DEFAULT (1) NOT NULL,  
    [DataCriacao] DATETIME2 (3) DEFAULT (sysutcdatetime()) NOT NULL,  
    CONSTRAINT [PK_tbProduto] PRIMARY KEY CLUSTERED ([ProdutoId] ASC)  
);  
  
CREATE TABLE [dbo].[tbOportunidade] (  
    [OportunidadeId] INT IDENTITY (1, 1) NOT NULL,  
    [ClienteId] INT NOT NULL,  
    [Titulo] VARCHAR (100) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,  
    [Descricao] NVARCHAR (MAX) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,  
    [Valor] DECIMAL (10, 2) NULL,
```

```

[DataFechamento]    DATE                NULL,
[Etapas]              VARCHAR (20)        COLLATE SQL_Latin1_General_CP1_CI_AS DEFAULT
('Prospecção') NOT NULL,
[Probabilidade]      INT                  NULL,
CONSTRAINT [PK_tbOportunidade] PRIMARY KEY CLUSTERED ([OportunidadeId] ASC),
CONSTRAINT [FK_tbOportunidade_tbCliente] FOREIGN KEY ([ClienteId]) REFERENCES
[dbo].[tbCliente] ([ClienteId]) ON DELETE NO ACTION,
CONSTRAINT [CK_tbOportunidade_Etapas] CHECK ([Etapas] IN
('Prospecção', 'Qualificação', 'Proposta', 'Negociação', 'Fechada')),
CONSTRAINT [CK_tbOportunidade_Probabilidade] CHECK ([Probabilidade] BETWEEN 0 AND
100)
);
CREATE TABLE [dbo].[tbAtividade] (
[AtividadeId]        INT                  IDENTITY (1, 1) NOT NULL,
[ClienteId]          INT                  NULL,
[OportunidadeId]     INT                  NULL,
[Tipo]                VARCHAR (20)        COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
[Descricao]           NVARCHAR (MAX)      COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[DataAgendada]        DATETIME2 (3)       NULL,
[DataConclusao]       DATETIME2 (3)       NULL,
[Status]              VARCHAR (20)        COLLATE SQL_Latin1_General_CP1_CI_AS DEFAULT
('Pendente') NOT NULL,
[Responsavel]         VARCHAR (100)       COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
CONSTRAINT [PK_tbAtividade] PRIMARY KEY CLUSTERED ([AtividadeId] ASC),
CONSTRAINT [FK_tbAtividade_tbCliente] FOREIGN KEY ([ClienteId]) REFERENCES
[dbo].[tbCliente] ([ClienteId]) ON DELETE SET NULL,
CONSTRAINT [FK_tbAtividade_tbOportunidade] FOREIGN KEY ([OportunidadeId])
REFERENCES [dbo].[tbOportunidade] ([OportunidadeId]) ON DELETE SET NULL,
CONSTRAINT [CK_tbAtividade_Tipo] CHECK ([Tipo] IN ('Ligação', 'E-
mail', 'Reunião', 'Tarefa')),
CONSTRAINT [CK_tbAtividade_Status] CHECK ([Status] IN
('Pendente', 'Concluída', 'Cancelada'))
);

```

● Ferramenta: Flyway Resultado:

O flyway OSS pode ser obtido e instalado do caminho

<https://repo1.maven.org/maven2/org/flywaydb/flyway-commandline/10.20.1/flyway-commandline-10.20.1-linux-x64.tar.gz>.

O flyway Enterprise foi obtido da url <https://www.red-gate.com/products/flyway/community/download/>.

O flyway requer a criação de um baseline no banco de dados com o comando:

```

$ flyway
  -url = $(JDBC_URL)
  -user = $(DB_USER)
  -password = $(DB_PASSWORD)
  -locations = $(FLYWAY_LOCATIONS)
baseline
  -baselineVersion = $(BASELINE_VERSION)
  -baselineDescription = "$(BASELINE_DESCRIPTION)"

```

```
Flyway OSS Edition 10.20.1 by Redgate

See release notes here: https://rd.gt/4160bM1
Database: jdbc:sqlserver://DESENV-SRVSQL.desenv.com\SQLCLUST06:53679;connectRetryInterval=10;connectRetryCount=1;maxResultBuffer=-1;sendTemporalDataTypesAsStringForBulkCopy=true;delayLoadingLobs=true;useFmtOnly=false;useBulkCopyForBatchInsert=false;cancelQueryTimeout=-1;sslProtocol=TLS;calcBigDecimalPrecision=false;useDefaultJaasConfig=false;jaasConfigurationName=SQLJDBCDriver;statementPoolingCacheSize=0;serverPreparedStatementDiscardThreshold=10;enablePrepareOnFirstPreparedStatementCall=false;flps=false;socketTimeout=0;authentication=NotSpecified;authenticationScheme=nativeAuthentication;xopenStates=false;datetimeParameterType=datetime2;sendTimeAsDatetime=true;replication=false;trustStoreType=JKS;trustServerCertificate=true;TransparentNetworkIPResolution=true;IPAddressPreference=IPv4First;serverNameAsACE=false;sendStringParametersAsUnicode=true;selectMethod=direct;responseBuffering=adaptive;queryTimeout=-1;packetSize=8000;multiSubnetFailover=false;loginTimeout=30;lockTimeout=-1;lastUpdateCount=true;useFlexibleCallableStatements=true;useDefaultGSSCredential=false;prepareMethod=prepxec;encrypt=false;disableStatementPooling=true;databaseName=OctopusSrvConexao;columnEncryptionSetting=Disabled;applicationName=Microsoft JDBC Driver for SQL Server;applicationIntent=readwrite; (Microsoft SQL Server 16.0)
Creating Schema History table [OctopusSrvConexao].[dbo].[flyway_schema_history] with baseline ...
Successfully baselined schema with version: 0
```

É criada a tabela flyway_schema_history na base de dados com o registro do *baseline*.

installed_rank	version	description	type	script	checksum	installed_by	installed_on	execution_time	success
1	0	Baseline	BASLINE	Baseline		gen_Octopus	2026-02-23 08:38:00.750	0	1

Artefatos do cenário

O flyway utiliza o formato de scripts V{Número de versao}__{descrição script}.sql para implantações e U{Número de versao}__{descrição script}.sql para rollbacks. Foram criado os scripts:

- V1__create_crm_tables.sql: junta todos CREATE de objetos no mesmo script.
- U1__create_crm_tables.sql: junta todos DROP de objetos no mesmo script.

Além disso foi criado o arquivo flyway.conf, que é opcional. O executável do flyway pode usar o arquivo de configuração ou parâmetros para a configuração da conexão de banco de dados e localização dos scripts. Abaixo o formato do arquivo.

```
flyway.url = jdbc:sqlserver://<host>:<port>;instanceName=<nome
instancia>;databaseName=<nome banco>;encrypt=true;trustServerCertificate=true
flyway.user = <nome usuario>
flyway.password = <senha do usuario>
flyway.schemas = <squema do banco>
flyway.locations = <localização dos scripts>
```

Implantando as tabelas

Para implantar basta executar o comando do flyway na mesma pasta que os scripts estão.

```
$ flyway
-url = $(JDBC_URL)
-user = $(DB_USER)
-password = $(DB_PASSWORD)
-locations = $(FLYWAY_LOCATIONS)
-target = 1
```

migrate

```
Flyway Enterprise Edition 12.0.2-rc1522 by Redgate
WARNING: You are using a limited Flyway trial license, valid until 2026-03-23. In 28 days you must either upgrade to a full Enterprise license or downgrade to Community.

See release notes here: https://rd.gt/4160bM1
Database: jdbc:sqlserver://DESENV-SRVSQL.desenv.com\SQLCLUST06:53679;connectRetryInterval=10;connectRetryCount=1;maxResultBuffer=-1;sendTemporalDataTypesAsStringForBulkCopy=true;delayLoadingLobs=true;useFmtOnly=false;cacheBulkCopyMetadata=false;bulkCopyForBatchInsertAllowInsecureValueModifications=false;bulkCopyForBatchInsertTableLock=false;bulkCopyForBatchInsertKeepNulls=false;bulkCopyForBatchInsertKeepIdentity=false;bulkCopyForBatchInsertFireTriggers=false;bulkCopyForBatchInsertCheckConstraints=false;bulkCopyForBatchInsertBatchSize=0;useBulkCopyForBatchInsert=false;cancelQueryTimeout=-1;sslProtocol=TLS;calcBigDecimalPrecision=false;useDefaultJaasConfig=false;jaasConfigurationName=SQLJDBCDriver;statementPoolingCacheSize=0;serverPreparedStatementDiscardThreshold=10;enablePrepareOnFirstPreparedStatementCall=false;flps=false;socketTimeout=0;authentication=NotSpecified;authenticationScheme=nativeAuthentication;xopenStates=false;datetimeParameterType=datetime2;sendTimeAsDatetime=true;replication=false;trustStoreType=JKS;trustServerCertificate=true;TransparentNetworkIPResolution=true;IPAddressPreference=IPv4First;serverNameAsACE=false;sendStringParametersAsUnicode=true;selectMethod=direct;responseBuffering=adaptive;queryTimeout=-1;packetSize=8000;multiSubnetFailover=false;loginTimeout=30;lockTimeout=-1;lastUpdateCount=true;useDefaultGSSCredential=false;prepareMethod=prepxec;encrypt=true;disableStatementPooling=true;databaseName=OctopusSrvConexao;columnEncryptionSetting=Disabled;applicationName=Microsoft JDBC Driver for SQL Server;applicationIntent=readwrite; (Microsoft SQL Server 16.0)
Successfully validated & migrated (execution time 00:00.046s)
Current version of schema [dbo]: 0
Migrating schema [dbo] to version "1 - create crm tables"
Successfully applied 1 migration to schema [dbo], now at version v1 (execution time 00:00.050s)
```

Desfazendo a implantação (rollback)

flyway -url="\$\$(JDBC_URL)" -user=\$(DB_USER) -password=\$(DB_PASSWORD) -locations=\$(FLYWAY_LOCATIONS) undo

```
Flyway Enterprise Edition 12.0.2-rc1522 by Redgate
WARNING: You are using a limited Flyway trial license, valid until 2026-03-23. In 28 days you must either upgrade to a full Enterprise license or downgrade to Community.

See release notes here: https://rd.gt/4160bM1
Database: jdbc:sqlserver://DESENV-SRVSQL.desenv.com\SQLCLUST06:53679;connectRetryInterval=10;connectRetryCount=1;maxResultBuffer=-1;sendTemporalDataTypesAsStringForBulkCopy=true;delayLoadingLobs=true;useFmtOnly=false;cacheBulkCopyMetadata=false;bulkCopyForBatchInsertAllowInsecureValueModifications=false;bulkCopyForBatchInsertTableLock=false;bulkCopyForBatchInsertKeepNulls=false;bulkCopyForBatchInsertKeepIdentity=false;bulkCopyForBatchInsertFireTriggers=false;bulkCopyForBatchInsertCheckConstraints=false;bulkCopyForBatchInsertBatchSize=0;useBulkCopyForBatchInsert=false;cancelQueryTimeout=-1;sslProtocol=TLS;calcBigDecimalPrecision=false;useDefaultJaasConfig=false;jaasConfigurationName=SQLJDBCDriver;statementPoolingCacheSize=0;serverPreparedStatementDiscardThreshold=10;enablePrepareOnFirstPreparedStatementCall=false;flps=false;socketTimeout=0;authentication=NotSpecified;authenticationScheme=nativeAuthentication;xopenStates=false;datetimeParameterType=datetime2;sendTimeAsDatetime=true;replication=false;trustStoreType=JKS;trustServerCertificate=true;TransparentNetworkIPResolution=true;IPAddressPreference=IPv4First;serverNameAsACE=false;sendStringParametersAsUnicode=true;selectMethod=direct;responseBuffering=adaptive;queryTimeout=-1;packetSize=8000;multiSubnetFailover=false;loginTimeout=30;lockTimeout=-1;lastUpdateCount=true;useDefaultGSSCredential=false;prepareMethod=prepxec;encrypt=true;disableStatementPooling=true;databaseName=OctopusSrvConexao;columnEncryptionSetting=Disabled;applicationName=Microsoft JDBC Driver for SQL Server;applicationIntent=readwrite; (Microsoft SQL Server 16.0)
Current version of schema [dbo]: 1
Undoing migration of schema [dbo] to version "1 - create crm tables"
Successfully undid 1 migration to schema [dbo], now at version v0 (execution time 00:00.196s)
```

Log das migrações

install d_rank	versio n	descripti on	type	script	checksum	installe d_by	installe d_on	execut ion_ti me	succe ss
1	0	Baseline	BASELINE	Baseline		gen_Octo pus	2026-02- 23 08:38:00 .750	0	1
2	1	create crm tables	SQL	V1__create_crm_table s.sql	188253142 3	gen_Octo pus	2026-02- 23 08:44:10 .083	48	1
3	2	drop crm tables	SQL	U1__create_crm_table s.sql	186959148 0	gen_Octo pus	2026-02- 23 08:47:12 .547	63	1

- **Observações:** O log exibe a versão atual e versão atualizada. Para os testes foi usado versão trial do enterprise, recurso de rollback só está disponível nas versões pagas.

- **Ferramenta: Liquibase Resultado:**

O liquibase foi instalado através do gerenciador de pacotes do sistema com “`apt-get install liquibase`”.

Artefatos do cenário

O liquibase permite usar formatos SQL, YAML, XML, JSON para definição das mudanças, ele organiza as mudanças em changesets onde comumente se coloca um objeto por vez, em cada changeset é definido o respectivo roolback da mudança. Ele permite organizar em vários arquivos ou um único arquivo. Para o teste foi usado o formato yaml, nele além dos scripts de mudança utilizamos um tipo específico que é o tagDatabase para fazer marcações de versão, iniciamos marcando uma versão 0.0 que é o estado inicial e ao final definimos a versão 1.0.

Segue abaixo o formato do arquivo **changelog-1.0.yaml** que utilizamos.

databaseChangeLog:

```
- changeSet:
  id: 0-tag-release-0-0
  author: codex
  changes:
    - tagDatabase:
      tag: 0.0
- changeSet:
  id: 1-create-tbCliente
  author: codex
  changes:
    - sql:
      dbms: mssql
      splitStatements: false
      stripComments: false
      sql: |
        <script create tbCliente>
  rollback:
    - sql:
      sql: |
        DROP TABLE [dbo].[tbCliente];
- changeSet:
  id: 2-create-tbProduto
  author: codex
  changes:
    - sql:
      dbms: mssql
      splitStatements: false
      stripComments: false
      sql: |
        <script create tbProduto>
  rollback:
```



```

- sql:
  sql: |
    DROP TABLE [dbo].[tbProduto];

- changeSet:
  id: 3-create-tbOportunidade
  author: codex
  changes:
    - sql:
      dbms: mssql
      splitStatements: false
      stripComments: false
      sql: |
        <script create tbOportunidade>
rollback:
  - sql:
    sql: |
      DROP TABLE [dbo].[tbOportunidade];

- changeSet:
  id: 4-create-tbContato
  author: codex
  changes:
    - sql:
      dbms: mssql
      splitStatements: false
      stripComments: false
      sql: |
        <script create tbcontato>
rollback:
  - sql:
    sql: |
      DROP TABLE [dbo].[tbContato];

- changeSet:
  id: 5-create-tbAtividade
  author: codex
  changes:
    - sql:
      dbms: mssql
      splitStatements: false
      stripComments: false
      sql: |
        <script create tbAtividade>
rollback:
  - sql:
    sql: |
      DROP TABLE [dbo].[tbAtividade];

- changeSet:
  id: 6-tag-release-1-0
  author: codex
  changes:
    - tagDatabase:
      tag: 1.0

```

Também criamos o arquivo de propriedades **liquibase.properties** conforme template abaixo.

```

changeLogFile = <caminho do yaml com mudanças>
url=jdbc:sqlserver://<host>:<porta>;instanceName=<instancia>;databaseName=<base>;encrypt=true;trustServerCertificate=true
username = <usuario>
password = <senha>
defaultSchemaName = <schema>
driver = com.microsoft.sqlserver.jdbc.SQLServerDriver
classpath = drivers/mssql-jdbc.jar
liquibase.hub.mode = off

```



```
logLevel = info
```

Implantação das tabelas

Comando para implantar tag 1.0

```
$ liquibase
```

```
--defaultsFile = liquibase.properties
--url = "$URL_CONNECTION"
--username = "$USER"
--password = "$PASSWORD"
--classpath = "$MSSQL_DRIVER_PATH"
```

```
updateToTag 1.0
```

```
Starting Liquibase at 12:17:06 using Java 21.0.9 (version 5.0.1 #9400 built at 2025-10-03 17:37+0000)
[2026-02-23 12:17:06] INFO [liquibase.ui] #####
##                                     ##
##      [Liquibase]                  ##
##                                     ##
##                                     ##
##                                     ##
##                                     ##
##                                     ##
## Get documentation at docs.liquibase.com ##
## Get certified courses at learn.liquibase.com ##
##                                     ##
#####
Starting Liquibase at 12:17:06 using Java 21.0.9 (version 5.0.1 #9400 built at 2025-10-03 17:37+0000)
Liquibase Version: 5.0.1
[2026-02-23 12:17:06] INFO [liquibase.ui] Liquibase Version: 5.0.1

[2026-02-23 12:17:06] INFO [liquibase.ui]
[2026-02-23 12:17:08] INFO [liquibase.integration] Starting command execution.
[2026-02-23 12:17:08] WARNING [liquibase.configuration] Potentially ignored key(s) in file exists at path liquibase.properties
- 'liquibase.hub.mode'
[2026-02-23 12:17:08] INFO [liquibase.snapshot] Creating snapshot
[2026-02-23 12:17:09] INFO [liquibase.lockservice] Successfully acquired change log lock
[2026-02-23 12:17:09] INFO [liquibase.changelog] Creating database changelog table with name: DATABASECHANGELOG
[2026-02-23 12:17:09] INFO [liquibase.changelog] Reading from DATABASECHANGELOG
[2026-02-23 12:17:09] INFO [liquibase.command] Using deploymentId: 1859826441
Running Changeset: poc1/changelog-1.0.yaml::1-create-tbCliente::codex
[2026-02-23 12:17:09] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::1-create-tbCliente::codex
[2026-02-23 12:17:09] INFO [liquibase.changelog] Custom SQL executed
[2026-02-23 12:17:09] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::1-create-tbCliente::codex ran successfully in 42ms
Running Changeset: poc1/changelog-1.0.yaml::2-create-tbProduto::codex
[2026-02-23 12:17:09] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::2-create-tbProduto::codex
[2026-02-23 12:17:09] INFO [liquibase.changelog] Custom SQL executed
[2026-02-23 12:17:09] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::2-create-tbProduto::codex ran successfully in 21ms
Running Changeset: poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex
[2026-02-23 12:17:09] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex
[2026-02-23 12:17:09] INFO [liquibase.changelog] Custom SQL executed
[2026-02-23 12:17:09] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex ran successfully in 25ms
Running Changeset: poc1/changelog-1.0.yaml::4-create-tbContato::codex
[2026-02-23 12:17:09] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::4-create-tbContato::codex
[2026-02-23 12:17:09] INFO [liquibase.changelog] Custom SQL executed
[2026-02-23 12:17:09] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::4-create-tbContato::codex ran successfully in 19ms
Running Changeset: poc1/changelog-1.0.yaml::5-create-tbAtividade::codex
[2026-02-23 12:17:09] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::5-create-tbAtividade::codex
[2026-02-23 12:17:09] INFO [liquibase.changelog] Custom SQL executed
[2026-02-23 12:17:09] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::5-create-tbAtividade::codex ran successfully in 26ms
Running Changeset: poc1/changelog-1.0.yaml::6-tag-release-1-0::codex
[2026-02-23 12:17:09] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::6-tag-release-1-0::codex
[2026-02-23 12:17:09] INFO [liquibase.changelog] Tag '1.0' applied to database
[2026-02-23 12:17:09] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::6-tag-release-1-0::codex ran successfully in 4ms
```

```

UPDATE SUMMARY
Run:                                6
Previously run:                     0
Filtered out:                       1
-----
Total change sets:                  7

FILTERED CHANGE SETS SUMMARY
After tag:                          1

[2026-02-23 12:17:09] INFO [liquibase.util] UPDATE SUMMARY
[2026-02-23 12:17:09] INFO [liquibase.util] Run:                                6
[2026-02-23 12:17:09] INFO [liquibase.util] Previously run:                     0
[2026-02-23 12:17:09] INFO [liquibase.util] Filtered out:                       1
[2026-02-23 12:17:09] INFO [liquibase.util] -----
[2026-02-23 12:17:09] INFO [liquibase.util] Total change sets:                  7
[2026-02-23 12:17:09] INFO [liquibase.util] FILTERED CHANGE SETS SUMMARY
[2026-02-23 12:17:09] INFO [liquibase.util] After tag:                          1
[2026-02-23 12:17:09] INFO [liquibase.util] Update summary generated
[2026-02-23 12:17:09] INFO [liquibase.command] Update command completed successfully.
Liquibase: Update-to-tag has been successful. Rows affected: 0
[2026-02-23 12:17:09] INFO [liquibase.ui] Liquibase: Update-to-tag has been successful. Rows affected: 0
[2026-02-23 12:17:10] INFO [liquibase.lockservice] Successfully released change log lock
[2026-02-23 12:17:10] INFO [liquibase.lockservice] Successfully released change log lock
[2026-02-23 12:17:10] INFO [liquibase.command] Command execution complete
Liquibase command 'updateToTag' was executed successfully.
[2026-02-23 12:17:11] INFO [liquibase.ui] Liquibase command 'updateToTag' was executed successfully.

```

Desfazendo criação (rollback)

Comando para desfazer última alteração.

```

$ liquibase
  --defaultsFile = liquibase.properties
  --url = "$URL_CONNECTION"
  --username = "$USER"
  --password = "$PASSWORD"
  --classpath = "$MSSQL_DRIVER_PATH"
rollback
  --tag = 0.0

```

```
Starting Liquibase at 12:36:41 using Java 21.0.9 (version 5.0.1 #9400 built at 2025-10-03 17:37+0000)
[2026-02-23 12:36:41] INFO [liquibase.ui] #####
##                               ##
##                               ##
##                               ##
##                               ##
##                               ##
##                               ##
##                               ##
##                               ##
##                               ##
##                               ##
## Get documentation at docs.liquibase.com    ##
## Get certified courses at learn.liquibase.com ##
##                               ##
#####
Starting Liquibase at 12:36:41 using Java 21.0.9 (version 5.0.1 #9400 built at 2025-10-03 17:37+0000)
Liquibase Version: 5.0.1
[2026-02-23 12:36:41] INFO [liquibase.ui] Liquibase Version: 5.0.1

[2026-02-23 12:36:41] INFO [liquibase.ui]
[2026-02-23 12:36:42] INFO [liquibase.integration] Starting command execution.
[2026-02-23 12:36:42] WARNING [liquibase.configuration] Potentially ignored key(s) in file exists at path liquibase.properties
- 'liquibase.hub.mode'
[2026-02-23 12:36:43] INFO [liquibase.changelog] Reading from DATABASECHANGELOG
[2026-02-23 12:36:44] INFO [liquibase.lockservice] Successfully acquired change log lock
[2026-02-23 12:36:44] INFO [liquibase.changelog] Reading from DATABASECHANGELOG
Rolling Back Changeset: poc1/changelog-1.0.yaml::6-tag-release-1-0::codex
[2026-02-23 12:36:44] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::6-tag-release-1-0::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::5-create-tbAtividade::codex
[2026-02-23 12:36:44] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::5-create-tbAtividade::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::4-create-tbContato::codex
[2026-02-23 12:36:44] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::4-create-tbContato::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex
[2026-02-23 12:36:44] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::2-create-tbProduto::codex
[2026-02-23 12:36:44] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::2-create-tbProduto::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::1-create-tbCliente::codex
[2026-02-23 12:36:44] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::1-create-tbCliente::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::0-tag-release-0-0::codex
[2026-02-23 12:36:44] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::0-tag-release-0-0::codex
[2026-02-23 12:36:44] INFO [liquibase.command] Rollback command completed successfully.
[2026-02-23 12:36:44] INFO [liquibase.lockservice] Successfully released change log lock
[2026-02-23 12:36:44] INFO [liquibase.command] Command execution complete
Liquibase command 'rollback' was executed successfully.
[2026-02-23 12:36:46] INFO [liquibase.ui] Liquibase command 'rollback' was executed successfully.
```

- **Observações:** O log de execução exibe quais arquivos/"*changeset*" executaram. Na primeira execução são criadas as tabelas DATABASECHANGELOGLOCK e DATABASECHANGELOG, sendo essa última responsável por guardar os registros dos changelog executados. O rollback remove os registros do changelog desfeitos. Na

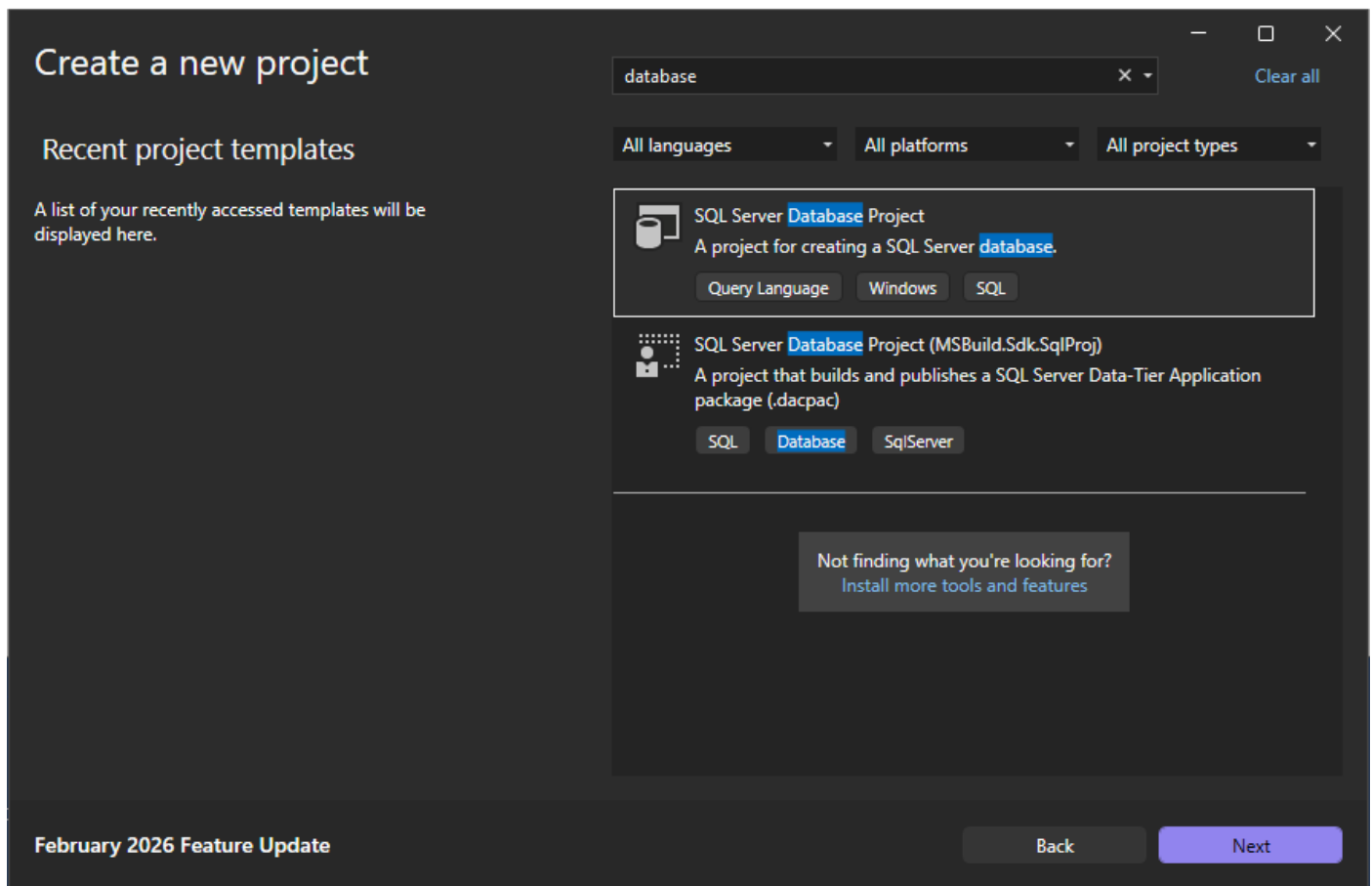
atualização do banco de dados é exibido um resumo das mudanças executadas e das que foram filtradas (não executadas).

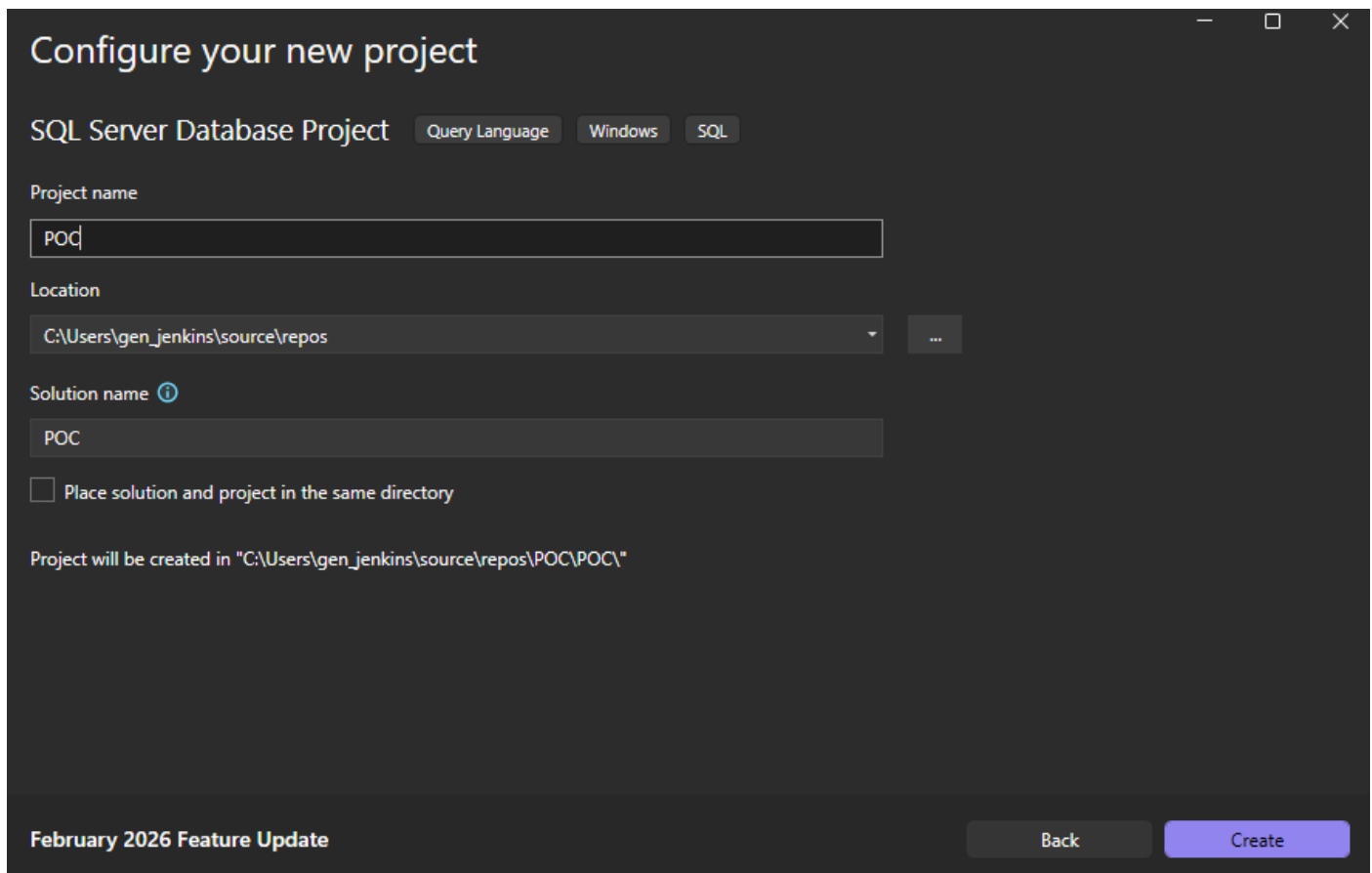
- **Ferramenta: SSDT Resultado:**

Criação do projeto

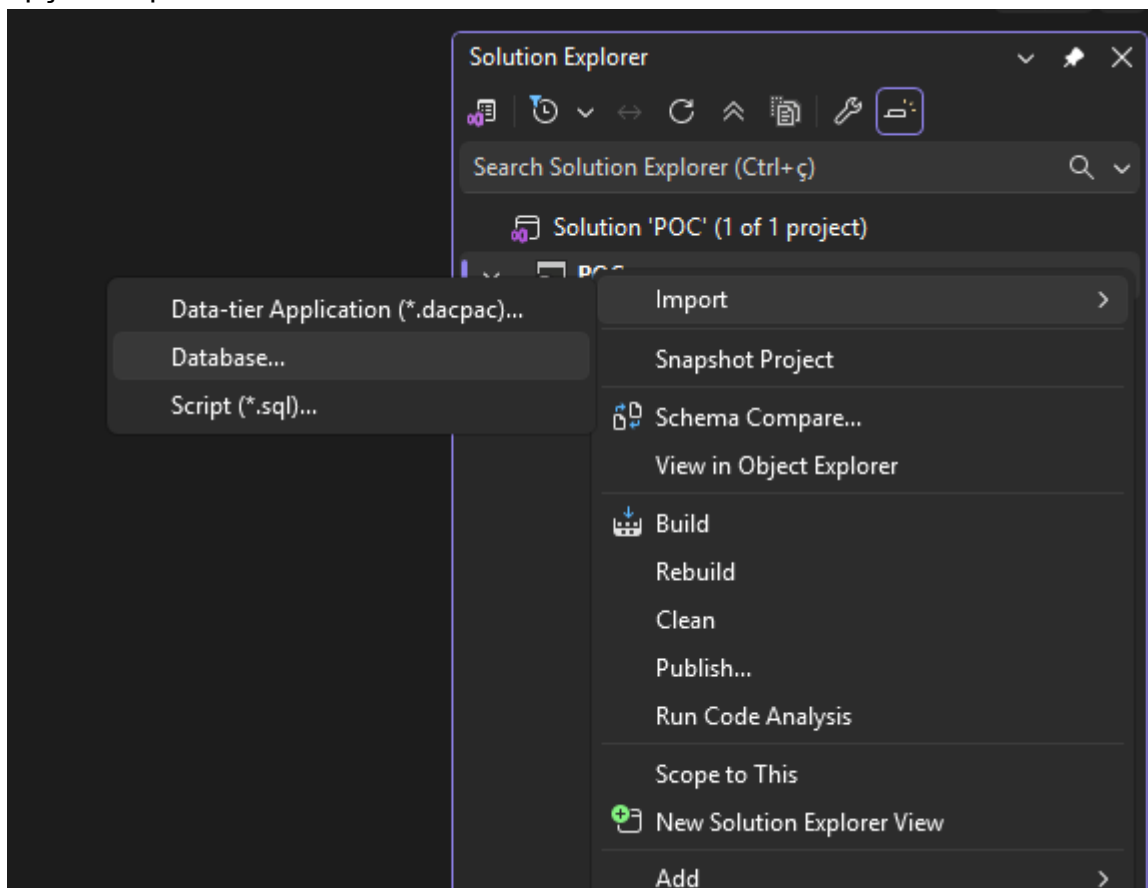
Pré-requisitos: Pacote SSDT instalado por meio do Visual Studio Instaler

Para criar com Visual Studio deve ser criado um projeto do tipo SQL Database Project

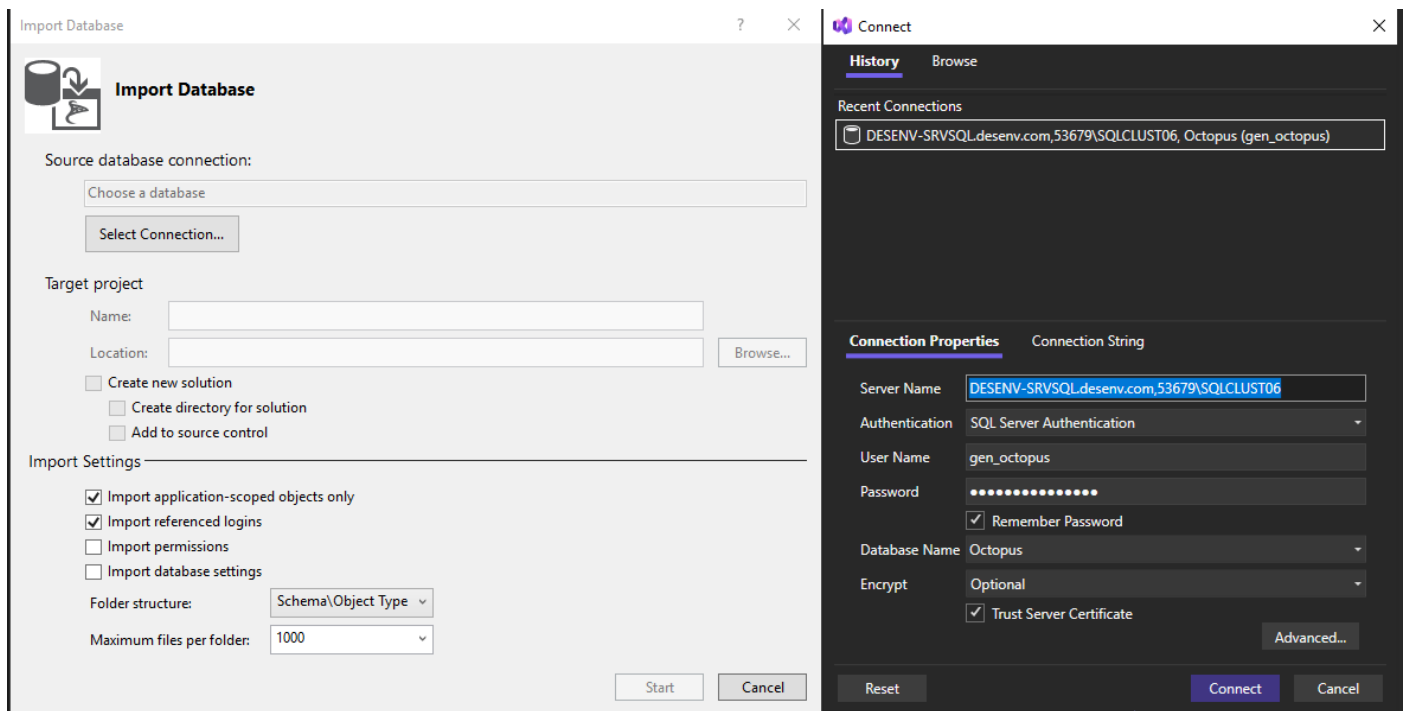




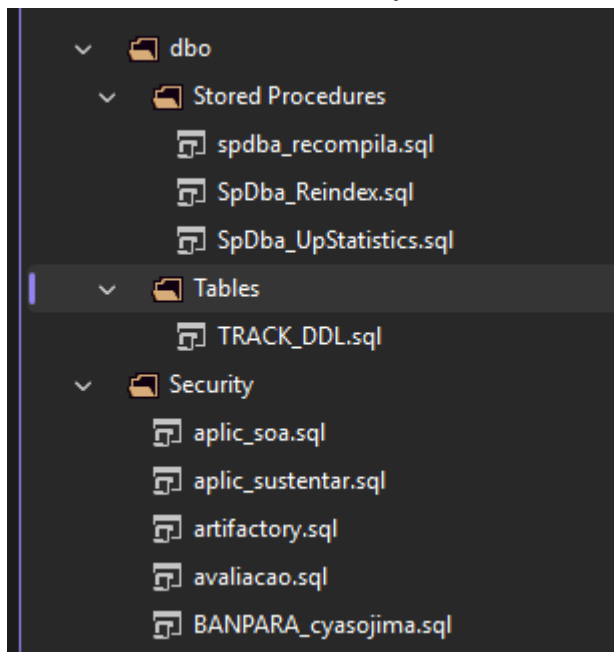
Após criado o projeto pode ser importada uma base de dados existente com direito e com a opção “Import”.



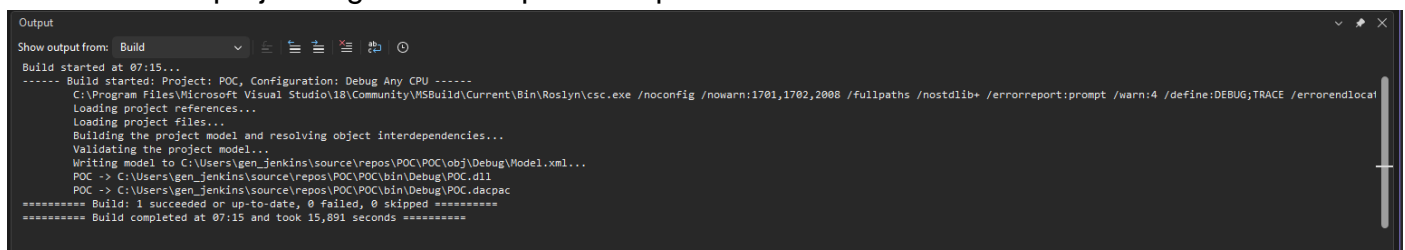
Deve ser configurada a conexão, depois configurar o escopo de objetos a ser importado e iniciar importação.

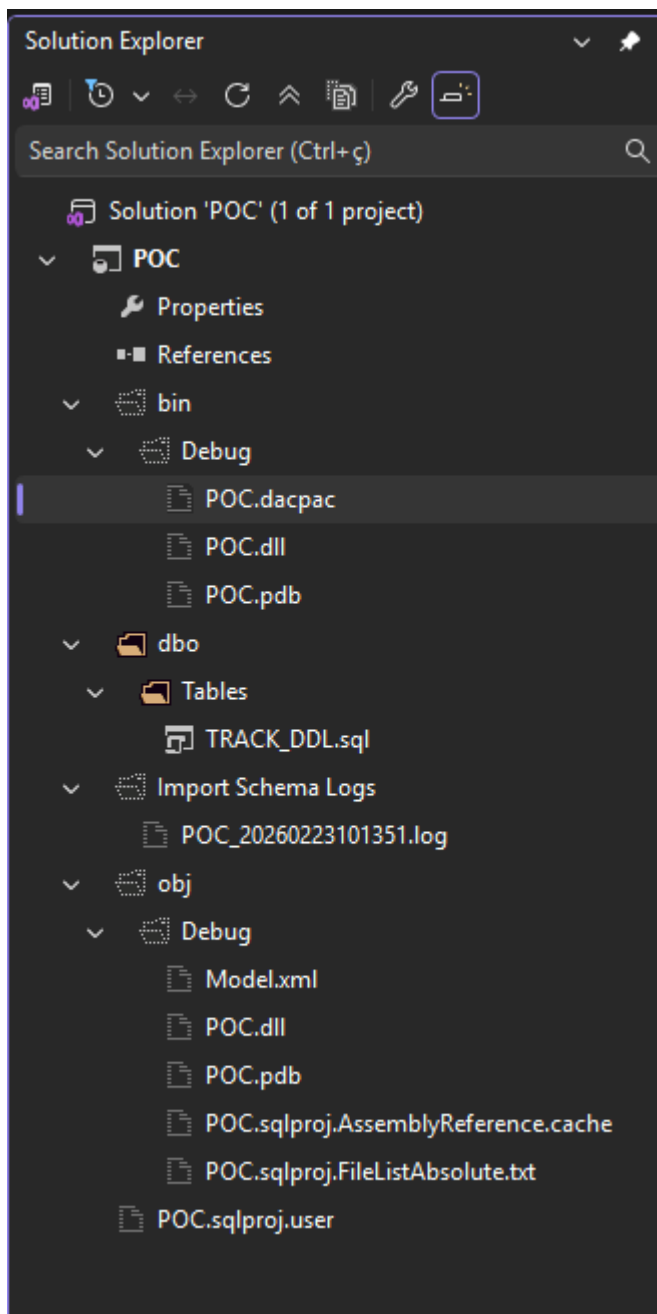


Após importado os objetos são carregados, em geral são carregadas algumas procedures default do SQL Server e objetos de usuário que podem ser removidos.



Ao construir o projeto é gerado o arquivo .dacpac.





No VSCode deve ser instalado primeiro a extensão SQL Database Projects.

EXTENSIONS: MARKETPLACE

SQL

SQL Database Projects
Enables users to develop and publis...
Microsoft

Prettier SQL VSCode
VSCode Extension to format SQL files
inferrinizzard

SQL Formatter
Format sql files using the sql-forma...
adpyke

SQLTools
Connecting users to many of the m...
Matheus Teixeira

SQLite
Explore and query SQLite databases.
alexvzz

Oracle SQL Develo...
End-to-end development tools for ...
Oracle Corporation

SQL Beautify
More Powerful extension that beaut...
clarkyu

Language PL/SQL
PL/SQL language (Oracle) support
xyz

SQLTools SQL Server
SQLTools SQL Server
Matheus Teixeira

Inline SQL
Highlight and lint SQL strings and S...
qufiwefwoyn

Extension: SQL Database Projects

SQL Database Projects
Microsoft | microsoft.com | 4,939,167 | ★★★★★ (5)
Enables users to develop and publish database schemas for MSSQL Databases

Disable | Uninstall | ☒ Auto Update

Prettier SQL VSCode v1.6.0
VSCode Extension to format SQL files
Hold Alt key to mouse over

SQL Database Projects extension

Database source control where objects are stored as code, integrating with build-time validation and dynamically calculated deployments for database DevOps. Cross-platform and compatible with Visual Studio SQL Server Data Tools (SSDT) for SQL Server, Azure SQL, and Fabric SQL.

Overview

SQL Database Projects for Azure Data Studio and VS Code provides a way to design, edit, and publish objects to SQL databases from a source controlled project. For a complete development workflow, build and deploy your database projects in CI/CD pipelines, such as [GitHub Actions](#) or Azure DevOps.

Workflow Diagram:

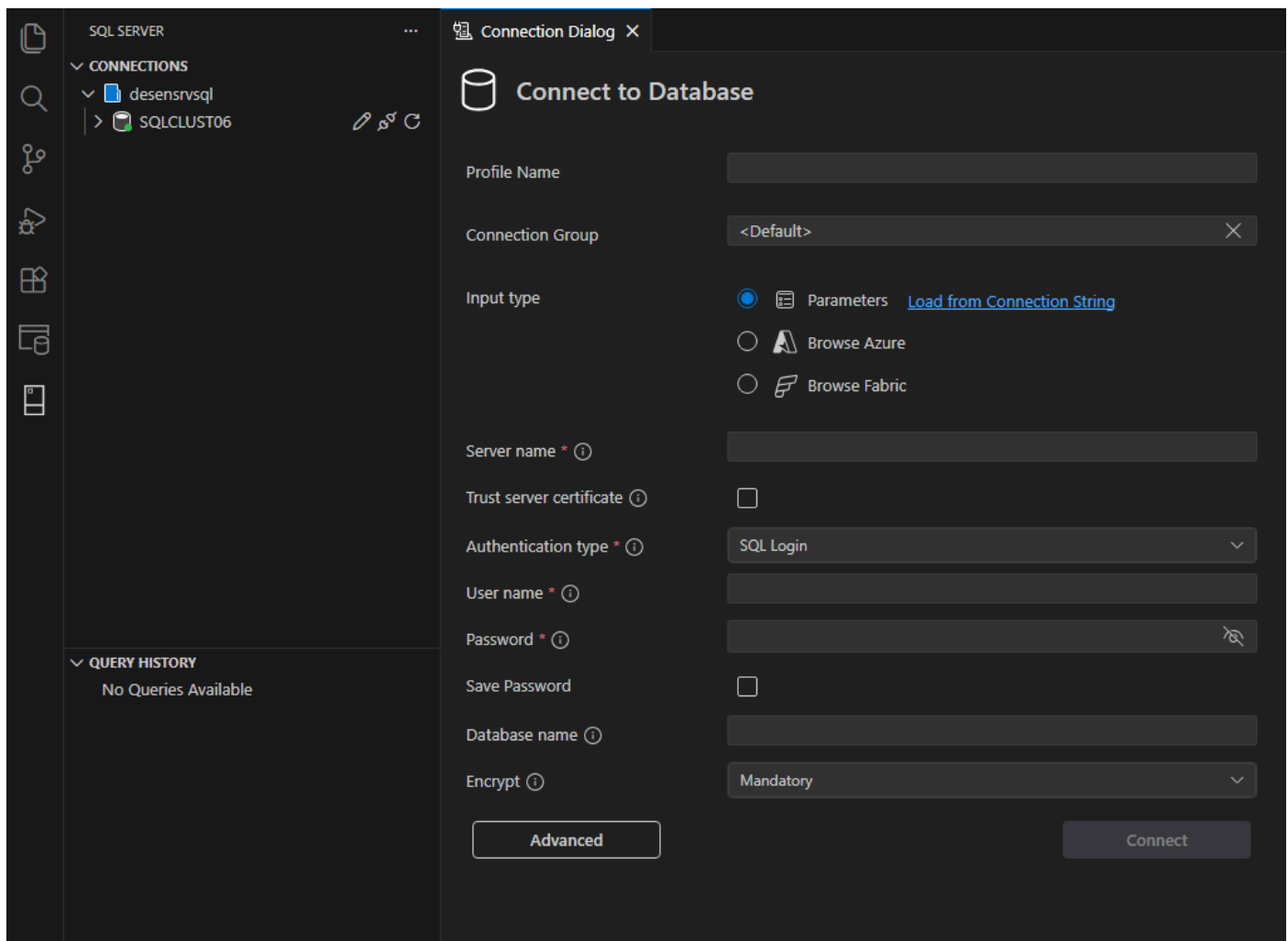
```
graph LR; A[.sqlproj] -- build --> B[.dacpac]; B -- deploy --> C[(Database)];
```

Learn more about SQL Database Projects in the documentation: <https://aka.ms/sqlprojects>

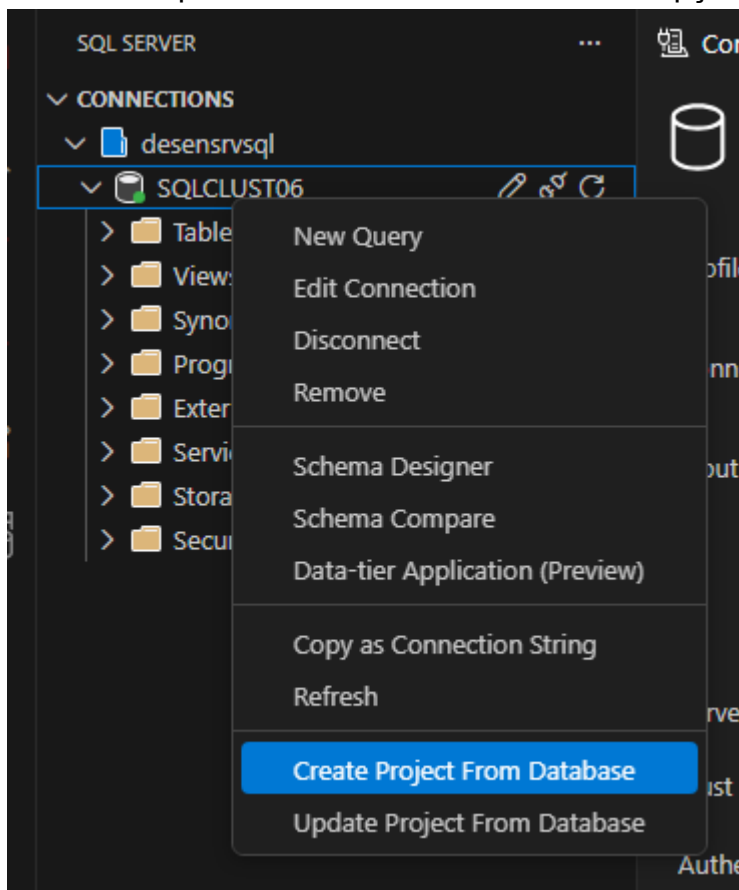
Requirements

The extension will prompt to install the .NET SDK if it is not found, as it is required to build SQL projects. You can also install the .NET SDK

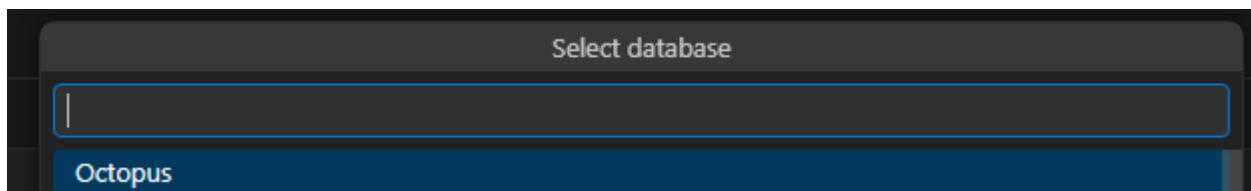
Com a extensão instalada deve ser configurada a conexão com banco de dados existente no caso de já existir um banco.



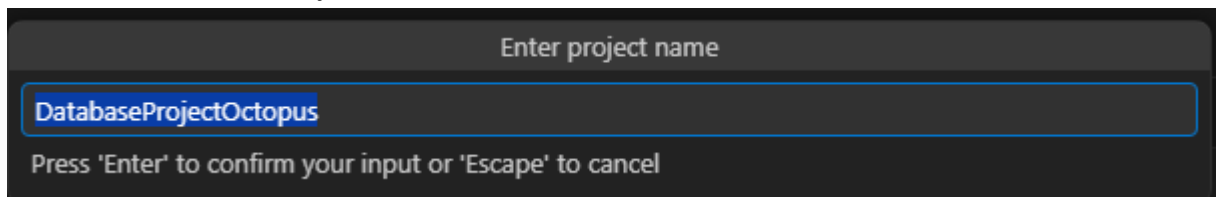
Com um clique direito na conexão escolher a opção “Create Project From Database”.



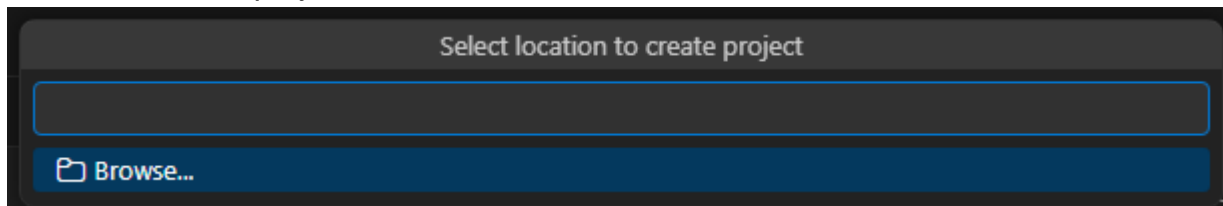
Escolher o banco de dados.



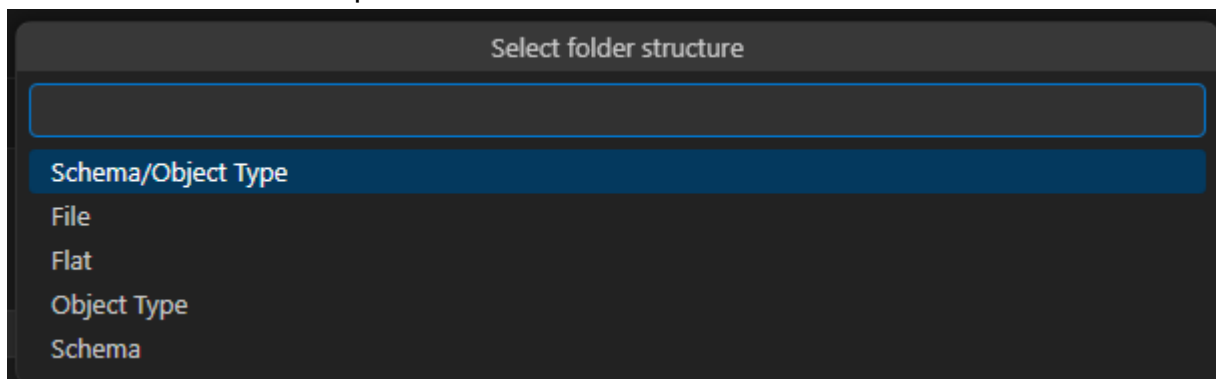
Definir o nome do Projeto.



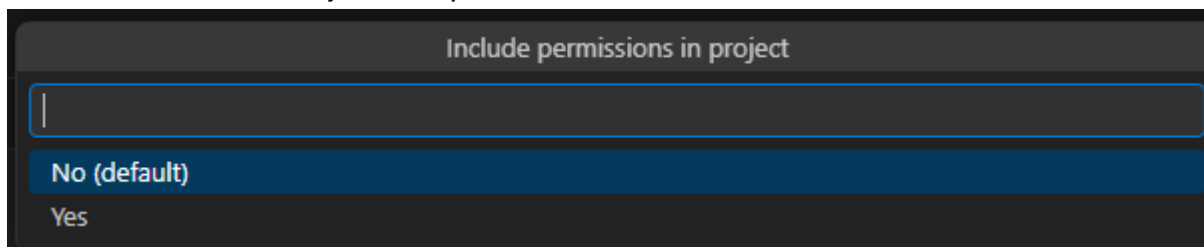
Definir o local do projeto



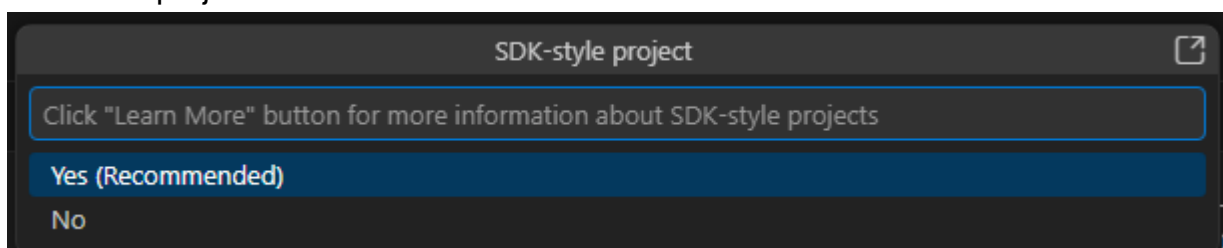
Definir a estrutura de arquivos



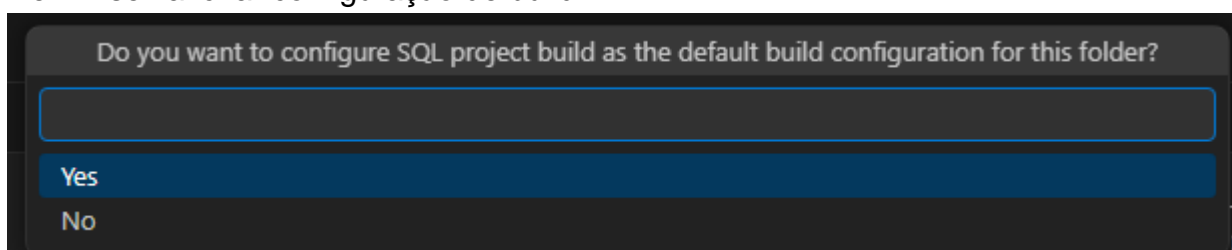
Definir se vai incluir objetos de permissão



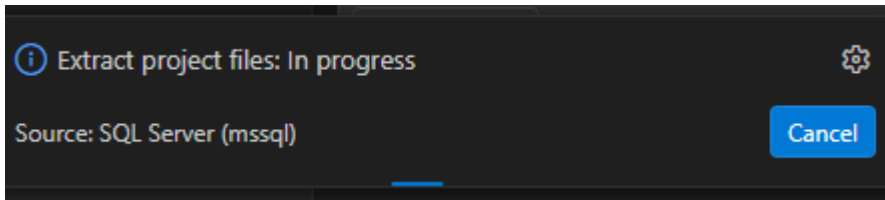
Definir o tipo de projeto, confirmando com sim será criado um projeto .net, caso contrário é criado um projeto .net framework.



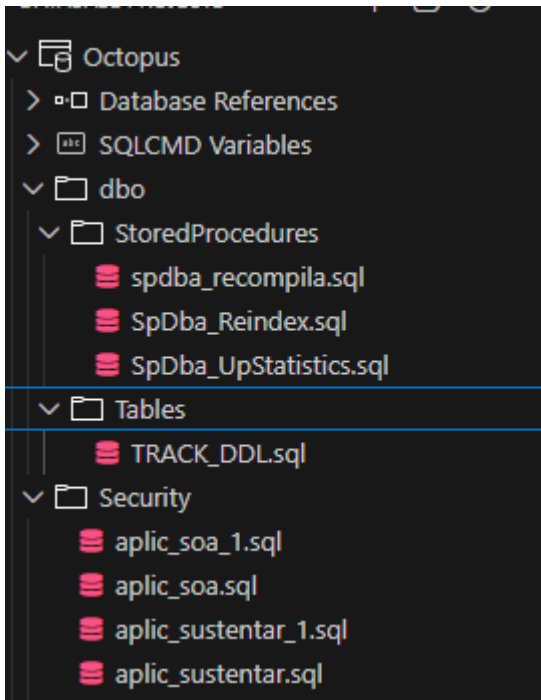
Definir se vai criar configuração de build



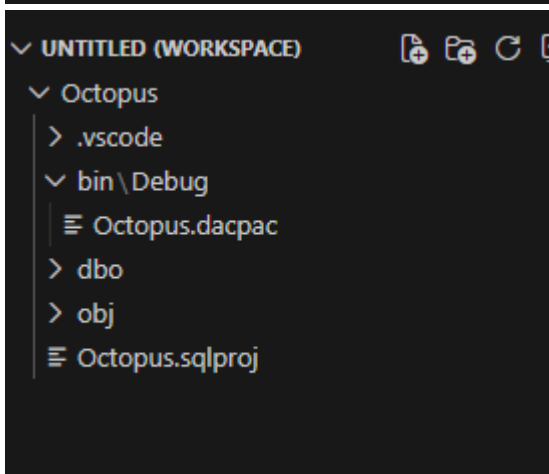
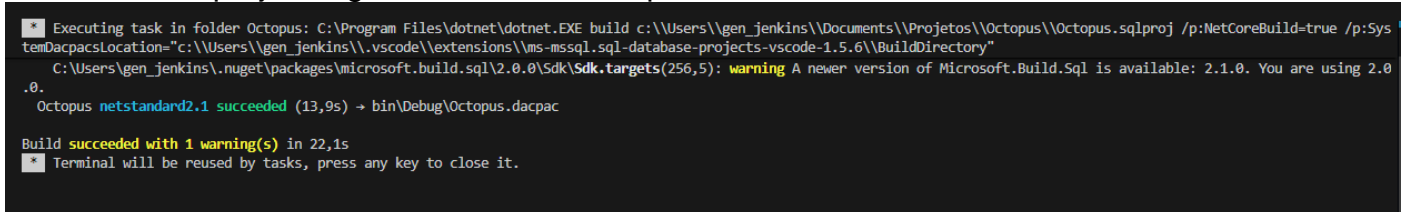
Após confirmar os objetos atuais da base vão ser extraídos e criados no projeto



Após finalizado os objetos são carregados, a pasta Security pode ser excluída para não alterar objetos de usuário, as procedures carregadas também são procedures defaults do SQL Server e podem ser removidas.

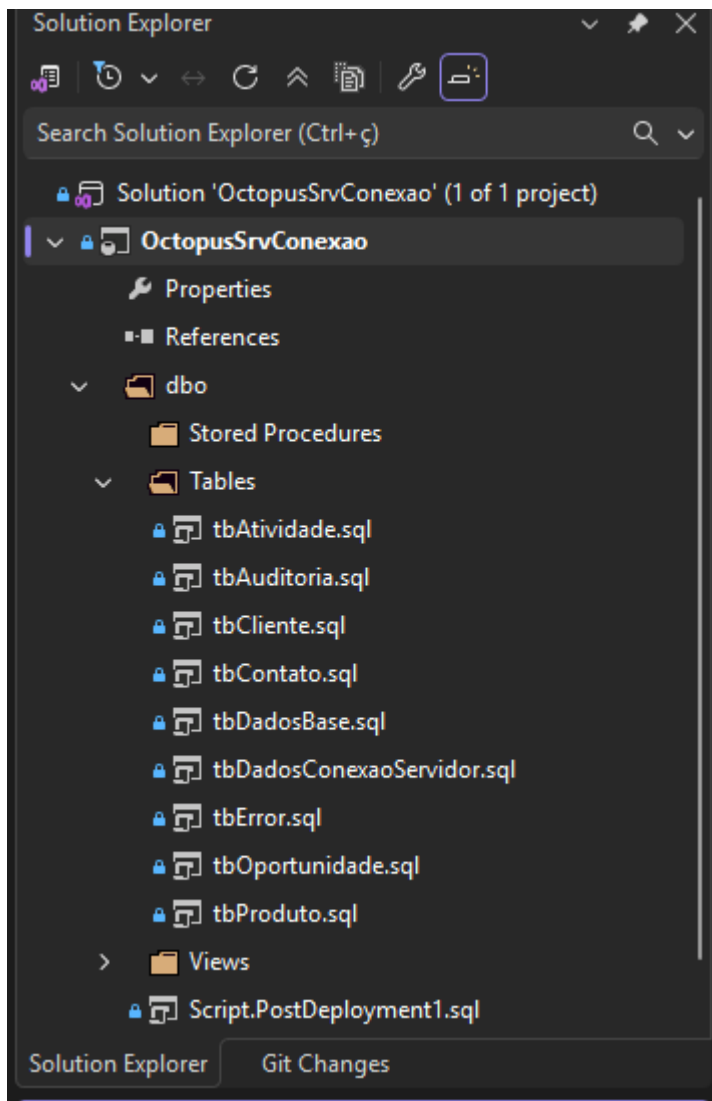


Construindo o projeto é gerado o binário .dacpac



Artefatos do cenário

O cenário usou uma base já existente e acrescentou as tabelas **tbAtividade**, **tbCliente**, **tbContato**, **tbOportunidade** e **tbProduto** conforme segue. Onde cada script corresponde ao respectivo Create table.



Construção do projeto com msbuild

```
$ msbuild OctopusSrvConexao.sqlproj
/p:Configuration = Release
/p:Platform = "Any CPU"
```

```
PS C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao> msbuild OctopusSrvConexao.sqlproj /p:Configuration=Release /p:Platform="Any CPU"
MSBuild version 17.4.0-18d5aef85 for .NET Framework
Build started 18/02/2026 16:56:32.
Project "C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao\OctopusSrvConexao.sqlproj" on node 1 (default targets).
GenerateSqlTargetFrameworkMoniker:
Skipping target "GenerateSqlTargetFrameworkMoniker" because all output files are up-to-date with respect to the input files.
SonarQubeImportBeforeInfo:
  Sonar: (OctopusSrvConexao) SonarQube.Integration.ImportBefore.targets was loaded
  Sonar: (OctopusSrvConexao.sqlproj) The analysis targets file not found: C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao\sonarqube\bin\targets\SonarQube.Integration.targets
CoreCompile:
  C:\Program Files\Microsoft Visual Studio\2022\Community\MSBuild\Current\Bin\Roslyn\csc.exe /noconfig /nowarn:1701,1702 /fullpaths /nostdlib+ /highentropyva+ /reference:"C:\Program Files (x86)\Reference Assemblies\Microsoft\Framework\.NETFramework\Version\v4.7.2\mscorlib.dll" /out:"obj\Any CPU\Release\OctopusSrvConexao.dll" /subsystemversion:6.00 /target:library /utf8output /langversion:7.3 "C:\Users\gen_jenkins\AppData\Local\Temp\3\*.NETFramework.Version\v4.7.2.SqlClrAttributes.cs"
CompilerServer: server - server processed compilation - 22b1dfc0-9e4b-41ad-9ec3-d15b1229debf
SqlBuild:
  Creating a model to represent the project...
  Loading project references...
  Loading project files...
  Building the project model and resolving object interdependencies...
  Validating the project model...
  Writing model to C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao\obj\Any CPU\Release\Model.xml...
CopyFilesToOutputDirectory:
  Copying file from "C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao\obj\Any CPU\Release\OctopusSrvConexao.dll" to "C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao\bin\Output\OctopusSrvConexao.dll"
  OctopusSrvConexao -> C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao\bin\Output\OctopusSrvConexao.dll
SqlPrepareForRun:
  OctopusSrvConexao -> C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao\bin\Output\OctopusSrvConexao.dacpac
Done Building Project "C:\Users\gen_jenkins\source\repos\OctopusSrvConexao\OctopusSrvConexao\OctopusSrvConexao.sqlproj" (default targets).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:19.16
```

Figura 1 Construção do projeto SSDT

Implantação da versão

```
$ sqlpackage.exe
/a:Publish
/sf:"<binfile>"
/tsn:"<server url>"
```

```
/tdn:"<nome_base>"
/tu:"<usuário>"
/tp:"<senha>"
/TargetTrustServerCertificate:true
```

```
Publishing to database 'OctopusSrvConexao' on server 'DESENV-SRVSQL.desenv.com,53679\SQLCLUST06'.
Initializing deployment (Start)
*** The object [OctopusSrvConexao] exists in the target, but it will not be dropped even though you selected the 'Generate drop statements for objects that are in the target database but that are not in the source' check box.
*** The object [OctopusSrvConexao_log] exists in the target, but it will not be dropped even though you selected the 'Generate drop statements for objects that are in the target database but that are not in the source' check box.
Initializing deployment (Complete)
Analyzing deployment plan (Start)
Analyzing deployment plan (Complete)
Updating database (Start)
Creating Table [dbo].[tbAtividade]...
Creating Table [dbo].[tbCliente]...
Creating Table [dbo].[tbContato]...
Creating Table [dbo].[tbOportunidade]...
Creating Table [dbo].[tbProduto]...
Creating Default Constraint unnamed constraint on [dbo].[tbAtividade]...
Creating Default Constraint unnamed constraint on [dbo].[tbCliente]...
Creating Default Constraint unnamed constraint on [dbo].[tbCliente]...
Creating Default Constraint unnamed constraint on [dbo].[tbContato]...
Creating Default Constraint unnamed constraint on [dbo].[tbOportunidade]...
Creating Default Constraint unnamed constraint on [dbo].[tbProduto]...
Creating Default Constraint unnamed constraint on [dbo].[tbProduto]...
Creating Foreign Key [dbo].[FK_tbAtividade_tbOportunidade]...
Creating Foreign Key [dbo].[FK_tbAtividade_tbCliente]...
Creating Foreign Key [dbo].[FK_tbContato_tbCliente]...
Creating Foreign Key [dbo].[FK_tbOportunidade_tbCliente]...
Creating Check Constraint [dbo].[CK_tbAtividade_Tipo]...
Creating Check Constraint [dbo].[CK_tbAtividade_Status]...
Creating Check Constraint [dbo].[CK_tbCliente_Status]...
Creating Check Constraint [dbo].[CK_tbOportunidade_Probabilidade]...
Creating Check Constraint [dbo].[CK_tbOportunidade_Etapa]...
Checking existing data against newly created constraints
Update complete.
Updating database (Complete)
Successfully published database.
Time elapsed 0:00:14.83
```

Rollback da versão (implantar build antigo)

```
$ sqlpackage.exe
/a:Publish
/sf:"<binfile>"
/tsn:"<server url>"
/tdn:"<nome_base>"
/tu:"<usuário>"
/tp:"<senha>"
/TargetTrustServerCertificate:true
/p:DropObjectsNotInSource=True
/p:BlockOnPossibleDataLoss=False
/p:ExcludeObjectTypes="Users;Logins;RoleMembership;Permissions"
```

```

Publishing to database 'OctopusSrvConexao' on server 'DESENV-SRVSQL.desenv.com,53679\SQLCLUST06'.
Initializing deployment (Start)
*** The object [OctopusSrvConexao] exists in the target, but it will not be dropped even though you selected the 'Generate drop statements for objects that are in the target database but that are not in the source' check box.
*** The object [OctopusSrvConexao_log] exists in the target, but it will not be dropped even though you selected the 'Generate drop statements for objects that are in the target database but that are not in the source' check box.
*** The table [dbo].[tbAtividade] is being dropped, data loss could occur.
*** The table [dbo].[tbCliente] is being dropped, data loss could occur.
*** The table [dbo].[tbContato] is being dropped, data loss could occur.
*** The table [dbo].[tbOportunidade] is being dropped, data loss could occur.
*** The table [dbo].[tbProduto] is being dropped, data loss could occur.
Initializing deployment (Complete)
Analyzing deployment plan (Start)
Analyzing deployment plan (Complete)
Updating database (Start)
Dropping Default Constraint unnamed constraint on [dbo].[tbAtividade]...
Dropping Default Constraint unnamed constraint on [dbo].[tbCliente]...
Dropping Default Constraint unnamed constraint on [dbo].[tbCliente]...
Dropping Default Constraint unnamed constraint on [dbo].[tbContato]...
Dropping Default Constraint unnamed constraint on [dbo].[tbOportunidade]...
Dropping Default Constraint unnamed constraint on [dbo].[tbProduto]...
Dropping Default Constraint unnamed constraint on [dbo].[tbProduto]...
Dropping Foreign Key [dbo].[FK_tbOportunidade_tbCliente]...
Dropping Foreign Key [dbo].[FK_tbContato_tbCliente]...
Dropping Foreign Key [dbo].[FK_tbAtividade_tbCliente]...
Dropping Foreign Key [dbo].[FK_tbAtividade_tbOportunidade]...
Dropping Check Constraint [dbo].[CK_tbAtividade_Status]...
Dropping Check Constraint [dbo].[CK_tbAtividade_Tipo]...
Dropping Check Constraint [dbo].[CK_tbCliente_Status]...
Dropping Check Constraint [dbo].[CK_tbOportunidade_Probabilidade]...
Dropping Check Constraint [dbo].[CK_tbOportunidade_Etapa]...
Dropping Table [dbo].[tbAtividade]...
Dropping Table [dbo].[tbCliente]...
Dropping Table [dbo].[tbContato]...
Dropping Table [dbo].[tbOportunidade]...
Dropping Table [dbo].[tbProduto]...
Update complete.
Updating database (Complete)
Successfully published database.
Time elapsed 0:00:14.03

```

- **Observações:** O log exibe todos objetos que foram criados/dropados. O DROP de objetos é bloqueado por segurança e precisa do parâmetro DropObjectsNotInSource=True para ser efetivado, do parâmetro BlockOnPossibleDataLoss=False para configurar se deve ignorar perda de dados e do parâmetro ExcludeObjectTypes para que sejam ignorados objetos de Usuário, Login e Permissões. Tanto a criação quanto rollback levaram cerca de 14 segundos.

Cenário 2: Alteração de Estrutura Existente

- **Descrição do teste:** Remover 1 coluna de uma tabela e adicionar uma coluna em outra tabela.
- **Artefatos afetados:** tbCliente, tbProduto

```

ALTER TABLE [dbo].[tbCliente]
ADD Bairro VARCHAR(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL;

```

```

ALTER TABLE [dbo].[tbProduto]
DROP COLUMN Categoria;

```

- **Ferramenta:** Flyway **Resultado:**

Artefatos do cenário

Para esse cenário foi criado o script **V2__change_crm_tables.sql** exatamente com as alterações dos objetos e o **U2__change_crm_tables.sql** que desfaz as mudanças (remove coluna Bairro da tabela tbCliente e recria a coluna Categoria na tabela tbProduto).

Implantando as tabelas

```
$ flyway
```



```
-url = $(JDBC_URL)
-user = $(DB_USER)
-password = $(DB_PASSWORD)
-locations = $(FLYWAY_LOCATIONS)
-target = 2
```

migrate

```
flyway Enterprise Edition 12.8.2-rc1522 by Redgate
WARNING: You are using a limited Flyway trial license, valid until 2026-03-23. In 28 days you must either upgrade to a full Enterprise license or downgrade to Community.

See release notes here: https://rd.gt/4160bWl
Database: jdbc:sqlserver://DESENV-SRVSQL.desenv.com\SQLCUST06:53679;connectRetryInterval=10;connectRetryCount=1;maxResultBuffer=-1;sendTemporalDataTypesAsStringForBulkCopy=true;delayLoadingLobs=true;useFmtOnly=false;cacheBulkCopyMetadata=false;bulkCopyForBatchInsertAllowEncryptedValueModifications=false;bulkCopyForBatchInsertTableLock=false;bulkCopyForBatchInsertKeepNulls=false;bulkCopyForBatchInsertKeepIdentity=false;bulkCopyForBatchInsertFireTriggers=false;bulkCopyForBatchInsertCheckConstraints=false;bulkCopyForBatchInsertBatchSize=0;useBulkCopyForBatchInsert=false;cancelQueryTimeout=-1;sslProtocol=TLS;calcBigDecimalPrecision=false;useDefaultJdbcConfig=false;jdbcConfigurationName=SQLJDBCDriver;statementPoolingCacheSize=0;serverPrepareThreshold=10;enablePrepareOnFirstPreparedStatementCall=false;flps=false;socketTimeout=0;authentication=NotSpecified;authenticationScheme=nativeAuthentication;xopenStates=false;dateTimeParameterType=dateTime2;sendTimeAsDatetime=true;replication=false;trustStoreType=JKS;trustServerCertificate=true;transparentNetworkIPResolution=true;IPAddressPreference=IPv4First;serverNameAsACE=false;sendStringParametersAsUnicode=true;selectMethod=direct;responseBuffering=adaptive;queryTimeout=-1;packetSize=8000;multiSubnetFailover=false;loginTimeout=30;lockTimeout=-1;lastUpdateCount=true;useDefaultSSCredentia=false;prepareMethod=preparedStatementPooling=true;databaseName=octopus-srv-conexao;columnEncryptionSetting=Disabled;applicationName=Microsoft JDBC Driver for SQL Server;applicationIntent=readwrite; (Microsoft SQL Server 16.0)
Successfully validated 10 migrations (execution time 00:00.036s)
Current version of schema [dbo]: 1
Migrating schema [dbo] to version "2 - change crm tables"
Successfully applied 1 migration to schema [dbo], now at version v2 (execution time 00:00.036s)
```

Desfazendo a implantação (rollback)

\$ flyway

```
-url = $(JDBC_URL)
-user = $(DB_USER)
-password = $(DB_PASSWORD)
-locations = $(FLYWAY_LOCATIONS)
```

undo

```
flyway Enterprise Edition 12.8.2-rc1522 by Redgate
WARNING: You are using a limited Flyway trial license, valid until 2026-03-23. In 28 days you must either upgrade to a full Enterprise license or downgrade to Community.

See release notes here: https://rd.gt/4160bWl
Database: jdbc:sqlserver://DESENV-SRVSQL.desenv.com\SQLCUST06:53679;connectRetryInterval=10;connectRetryCount=1;maxResultBuffer=-1;sendTemporalDataTypesAsStringForBulkCopy=true;delayLoadingLobs=true;useFmtOnly=false;cacheBulkCopyMetadata=false;bulkCopyForBatchInsertAllowEncryptedValueModifications=false;bulkCopyForBatchInsertTableLock=false;bulkCopyForBatchInsertKeepNulls=false;bulkCopyForBatchInsertKeepIdentity=false;bulkCopyForBatchInsertFireTriggers=false;bulkCopyForBatchInsertCheckConstraints=false;bulkCopyForBatchInsertBatchSize=0;useBulkCopyForBatchInsert=false;cancelQueryTimeout=-1;sslProtocol=TLS;calcBigDecimalPrecision=false;useDefaultJdbcConfig=false;jdbcConfigurationName=SQLJDBCDriver;statementPoolingCacheSize=0;serverPrepareThreshold=10;enablePrepareOnFirstPreparedStatementCall=false;flps=false;socketTimeout=0;authentication=NotSpecified;authenticationScheme=nativeAuthentication;xopenStates=false;dateTimeParameterType=dateTime2;sendTimeAsDatetime=true;replication=false;trustStoreType=JKS;trustServerCertificate=true;transparentNetworkIPResolution=true;IPAddressPreference=IPv4First;serverNameAsACE=false;sendStringParametersAsUnicode=true;selectMethod=direct;responseBuffering=adaptive;queryTimeout=-1;packetSize=8000;multiSubnetFailover=false;loginTimeout=30;lockTimeout=-1;lastUpdateCount=true;useDefaultSSCredentia=false;prepareMethod=preparedStatementPooling=true;databaseName=octopus-srv-conexao;columnEncryptionSetting=Disabled;applicationName=Microsoft JDBC Driver for SQL Server;applicationIntent=readwrite; (Microsoft SQL Server 16.0)
Successfully validated 10 migrations (execution time 00:00.036s)
Current version of schema [dbo]: 1
Migrating schema [dbo] to version "2 - change crm tables"
Successfully applied 1 migration to schema [dbo], now at version v2 (execution time 00:00.036s)
```

Log de versões:

INSTALLED_RANK	VERSION	DESCRIPTION	TYPE	SCRIPT	CHECKSUM	INSTALLED_BY	INSTALLED_ON	EXECUTION_TIME	SUCCESS
1	0	Baseline	BASELINE	Baseline		gen_Octopus	2026-02-23 08:51:38.213	0	1
2	1	create crm tables	SQL	V1__create_crm_tables.sql	1882531423	gen_Octopus	2026-02-23 08:52:36.877	60	1
3	1	create crm tables	UNDO_SQL	U1__create_crm_tables.sql	1869591480	gen_Octopus	2026-02-23 09:52:48.997	61	1
4	1	create crm tables	SQL	V1__create_crm_tables.sql	1882531423	gen_Octopus	2026-02-23 09:53:18.617	59	1
5	2	change crm tables	SQL	V2__change_crm_tables.sql	97879311	gen_Octopus	2026-02-23 11:42:48.870	36	1
6	2	change crm tables	UNDO_SQL	U2__change_crm_tables.sql	1054429060	gen_Octopus	2026-02-23 11:45:38.070	79	

- **Observações:** Os scripts de alteração não exibem o objeto como um todo, apenas as mudanças incrementais. O log de versões especifica exatamente os scripts que foram executados

● Ferramenta: Liquibase Resultado:

Artefatos do cenário

Para esse cenário foram acrescentados os changeset no yaml **changelog-1.0.yml**.

- **changeSet:**
 - id: 7-change-crm-tables
 - author: codex
 - changes:
 - sql:

```

    dbms: mssql
    splitStatements: false
    stripComments: false
    sql: |
        <script de alteração>
rollback:
- sql:
    sql: |
        ALTER TABLE [dbo].[tbCliente]
        DROP COLUMN Bairro;

        ALTER TABLE [dbo].[tbProduto]
        ADD Categoria VARCHAR (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL;

- changeSet:
    id: 8-tag-release-2-0
    author: codex
    changes:
        - tagDatabase:
            tag: 2.0

```

Implantação das tabelas

Comando para implantar tag 1.0

```

$ liquibase
--defaultsFile=liquibase.properties
--url = "$URL_CONNECTION"
--username = "$USER"
--password = "$PASSWORD"
--classpath = "$MSSQL_DRIVER_PATH"
updateToTag 2.0

```



```

Publishing to database 'OctopusSrvConexao' on server 'DESENV-SRVSQL.desenv.com,53679\SQLCLUST06'.
Initializing deployment (Start)
*** The object [OctopusSrvConexao] exists in the target, but it will not be dropped even though you selected the 'Generate drop statements for objects that are in the target database but that are not in the source' check box.
*** The object [OctopusSrvConexao_log] exists in the target, but it will not be dropped even though you selected the 'Generate drop statements for objects that are in the target database but that are not in the source' check box.
*** The column [dbo].[tbProduto].[Categoria] is being dropped, data loss could occur.
Initializing deployment (Complete)
Analyzing deployment plan (Start)
Analyzing deployment plan (Complete)
Updating database (Start)
Dropping Default Constraint unnamed constraint on [dbo].[tbOportunidade]...
Dropping Check Constraint [dbo].[CK_tbAtividade_Status]...
Dropping Check Constraint [dbo].[CK_tbAtividade_Tipo]...
Dropping Check Constraint [dbo].[CK_tbCliente_Status]...
Dropping Check Constraint [dbo].[CK_tbOportunidade_Etapa]...
Dropping Check Constraint [dbo].[CK_tbOportunidade_Probabilidade]...
Altering Table [dbo].[tbCliente]...
Altering Table [dbo].[tbProduto]...
Creating Default Constraint unnamed constraint on [dbo].[tbOportunidade]...
Creating Check Constraint [dbo].[CK_tbAtividade_Status]...
Creating Check Constraint [dbo].[CK_tbAtividade_Tipo]...
Creating Check Constraint [dbo].[CK_tbCliente_Status]...
Creating Check Constraint [dbo].[CK_tbOportunidade_Etapa]...
Creating Check Constraint [dbo].[CK_tbOportunidade_Probabilidade]...
Checking existing data against newly created constraints
Update complete.
Updating database (Complete)
Successfully published database.
Time elapsed 0:00:10.30

```

Desfazendo alterações (Rollback)

```

Publishing to database 'OctopusSrvConexao' on server 'DESENV-SRVSQL.desenv.com,53679\SQLCLUST06'.
Initializing deployment (Start)
*** The object [OctopusSrvConexao] exists in the target, but it will not be dropped even though you selected the 'Generate drop statements for objects that are in the target database but that are not in the source' check box.
*** The object [OctopusSrvConexao_log] exists in the target, but it will not be dropped even though you selected the 'Generate drop statements for objects that are in the target database but that are not in the source' check box.
*** The column [dbo].[tbCliente].[Bairro] is being dropped, data loss could occur.
Initializing deployment (Complete)
Analyzing deployment plan (Start)
Analyzing deployment plan (Complete)
Updating database (Start)
Dropping Default Constraint unnamed constraint on [dbo].[tbProduto]...
Dropping Default Constraint unnamed constraint on [dbo].[tbProduto]...
Dropping Default Constraint unnamed constraint on [dbo].[tbOportunidade]...
Dropping Check Constraint [dbo].[CK_tbAtividade_Status]...
Dropping Check Constraint [dbo].[CK_tbAtividade_Tipo]...
Dropping Check Constraint [dbo].[CK_tbCliente_Status]...
Dropping Check Constraint [dbo].[CK_tbOportunidade_Etapa]...
Dropping Check Constraint [dbo].[CK_tbOportunidade_Probabilidade]...
Altering Table [dbo].[tbCliente]...
Starting rebuilding table [dbo].[tbProduto]...
Caution: Changing any part of an object name could break scripts and stored procedures.
Caution: Changing any part of an object name could break scripts and stored procedures.
Creating Default Constraint unnamed constraint on [dbo].[tbOportunidade]...
Creating Check Constraint [dbo].[CK_tbAtividade_Status]...
Creating Check Constraint [dbo].[CK_tbAtividade_Tipo]...
Creating Check Constraint [dbo].[CK_tbCliente_Status]...
Creating Check Constraint [dbo].[CK_tbOportunidade_Etapa]...
Creating Check Constraint [dbo].[CK_tbOportunidade_Probabilidade]...
Checking existing data against newly created constraints
Update complete.
Updating database (Complete)
Successfully published database.
Time elapsed 0:00:10.19

```

- **Observações:** Os objetos atualizados aparecem no log com “Altering Table” e “Starting rebuild table”. O parâmetro DropObjectsNotInSource=True faz com que as constraints sejam recriadas, o parâmetro DoNotDropObjectTypes="CheckConstraints;DefaultConstraints" pode ser usado para evitar esse comportamento.

Cenário 3: Integração com Pipeline CI/CD

- **Descrição do teste:** Adaptar ou configurar agentes para poder executar build/deploy, criar pipelines de teste de implantação e rollback.
- **Ferramenta: Flyway Resultado:**

Para configurar o flyway foi feito upload no nexus do arquivo .tar.gz e adicionado no Dockerfile do agente banpara-base o seguinte trecho.

```
RUN wget --no-check-certificate --user nexus-reader --password nexus-reader
https://nexus.banpara.com/repository/raw-release/FlywayDesktopLinuxX64_9.1.0.0.tar.gz -
P /tmp && \
    tar -xzf /tmp/FlywayDesktopLinuxX64_9.1.0.0.tar.gz -C /opt ./flyway/ && \
    rm -f /tmp/FlywayDesktopLinuxX64_9.1.0.0.tar.gz && \
    chown -R jenkins:jenkins /opt/flyway && \
    ln -s /opt/flyway/flyway /usr/local/bin
```

O pipeline do flyway é o seguinte:

```
@Library('banpara-esteiras') _

pipeline {
    agent {
        label 'banpara-base'
    }

    options {
        buildDiscarder(logRotator(numToKeepStr: '10'))
        ansiColor('xterm')
    }

    parameters {
        string(
            name: 'TARGET_VERSION',
            defaultValue: '1',
            description: 'Versão alvo para migração (ex: 1, 2, 3...)'
        )
        booleanParam(
            name: 'BASELINE',
            defaultValue: false,
            description: 'Executar flyway baseline antes da migração'
        )
        booleanParam(
            name: 'UNDO',
            defaultValue: false,
            description: 'Executar flyway undo (rollback)'
        )
    }

    environment {
        MSSQL_DRIVER_PATH = '/opt/flyway/drivers/mssql-jdbc-12.10.2.jre11.jar'
        FLYWAY_BIN = 'flyway'
        DB_HOST = 'DESENV-SRVSQLE.desenv.com'
        DB_PORT = '53679'
        DB_INSTANCE = 'SQLCLUST06'
        DB_NAME = 'OctopusSrvConexao'
        DB_ENCRYPT = 'true'
        DB_TRUST_SERVER_CERT = 'true'
        FLYWAY_LOCATIONS = "filesystem:${WORKSPACE}/flyway"
        BASELINE_VERSION = '0'
        BASELINE_DESCRIPTION = 'Initial Baseline'
        REDGATE_LICENSING_PERMIT = '${LICENSING_PERMIT}'
    }
}
```

```

}

stages {
    stage('Checkout') {
        steps {
            checkout([
                $class: 'GitSCM',
                branches: [[name: 'principal']],
                extensions: [],
                gitTool: 'Default',
                userRemoteConfigs: [[
                    credentialsId: 'ldap.gen_jenkins.credenciais',
                    url:
'https://gitlab.banpara.com/tfoliveira/octopussrvconexao.git'
                ]]
            ])
        }
    }

    stage('Executar Flyway') {
        steps {
            script {
                withCredentials([usernamePassword(
                    credentialsId: 'desensrvsqlsqlclust06.gen_octopus.credenciais',
                    usernameVariable: 'DB_USER',
                    passwordVariable: 'DB_PASSWORD'
                )]) {
                    dir('flyway') {
                        def commonFlags = "-
url=\"jdbc:sqlserver://${DB_HOST}:${DB_PORT};instanceName=${DB_INSTANCE};databaseName=${
DB_NAME};encrypt=${DB_ENCRYPT};trustServerCertificate=${DB_TRUST_SERVER_CERT}\" " +
                        "-user=${DB_USER} " +
                        "-password=${DB_PASSWORD} " +
                        "-locations=filesystem:${WORKSPACE}/flyway"

                        if (params.BASELINE) {
                            sh """
                                ${FLYWAY_BIN} ${commonFlags} baseline -
baselineVersion=${BASELINE_VERSION} -baselineDescription=\"${BASELINE_DESCRIPTION}\"
                                """
                        }

                        if (params.UNDO) {
                            sh """
                                ${FLYWAY_BIN} ${commonFlags} undo
                                """
                        } else {
                            sh """
                                ${FLYWAY_BIN} ${commonFlags} -
target=${params.TARGET_VERSION} migrate
                                """
                        }
                    }
                }
            }
        }
    }
}

```



```

    }
}

post {
    always {
        cleanWs()
    }
}
}
}

```

O pipeline baixa o repositório com scripts e executa o comando do flyway. Abaixo alguns logs das implantações. Ele foi criado com os parâmetros de versão, número a ser implantado, Baseline para indicar se é necessário criar baseline do flyway (cria tabela necessária para o flyway) e undo que indica se deve ser feito rollback.

Pipeline poc-flyway

Esta construção requer parâmetros:

TARGET_VERSION

Versão alvo para migração (ex: 1, 2, 3...)

☐ **BASELINE**
 Executar flyway baseline antes da migração

☐ **UNDO**
 Executar flyway undo (rollback)

▶ Construir
Cancel

Painel de controle > Sandbox > Thiago Oliveira > poc-flyway > #3

```

+ flyway '-url=jdbc:sqlserver://DESEW-SRVSQL.desenv.com:53679;instanceName=SQLCLUST06;databaseName=OctopusSrvConexao;encrypt=true;trustServerCertificate=true' -user=gen_octopus -password=**** -
locations=filesystem:/home/jenkins/agent/workspace/sandbox/Thiago-Oliveira/poc-flyway/flyway -target=1 migrate
Picked up JAVA_TOOL_OPTIONS: -Xms1g -Xmx4g -Dhttps.proxyHost=10.0.1.186 -Dhttps.proxyPort=8080 -Dhttps.proxyUser=serviceunix -Dhttps.proxyPassword=Ab123456 -
Dhttp.nonProxyHosts="*.banpara.com|*.homologacao.com|*.desenv.com|10.*.*.*|192.168.*.*|srpserverbi"
Flyway Community Edition 12.0.2-rc1522 by Redgate

See release notes here: https://rd.gt/4160bNI
Database: jdbc:sqlserver://DESEW-
SRVSQL.desenv.com:SQLCLUST06:53679;connectRetryInterval=10;connectRetryCount=1;maxResultBuffer=-1;sendTemporalDataTypesAsStringForBulkCopy=true;delayLoadingObs=true;useMtlOnly=false;cacheBulkCopyMetadata=false;bulk
CopyForBatchInsertAllowEncryptedValueModifications=false;bulkCopyForBatchInsertTableLock=false;bulkCopyForBatchInsertKeepNulls=false;bulkCopyForBatchInsertKeepIdentity=false;bulkCopyForBatchInsertFireTriggers=false;
bulkCopyForBatchInsertCheckConstraints=false;bulkCopyForBatchInsertBatchSize=0;useBulkCopyForBatchInsert=false;cancelQueryTimeout=-1;sslProtocol=TLS;calBigDecimalPrecision=false;useDefaultJaaSConfig=false;jaasConfi
gurationName=SQLJDBCDriver;statementPoolingCacheSize=0;serverPreparedStatementDiscardThreshold=10;enablePrepareOnFirstPreparedStatementCall=false;fips=false;socketTimeout=0;authentication=NotSpecified;authentication
Scheme=NativeAuthentication;openStates=false;dateTimeParameterType=dateTime;sendTimeAsDateTime=true;replication=false;trustStoreType=JKS;trustServerCertificate=true;transparentNetworkIPResolution=true;IPAddressPre
ference=IPv4First;serverNameAsACE=false;sendStringParametersAsUnicode=true;selectMethod=direct;responseBuffering=adaptive;queryTimeout=-1;packetSize=8000;multiSubnetFailover=false;loginTimeout=30;lockTimeout=-1;last
UpdateCount=true;useDefaultTSSCredential=false;prepareMethod=prepare;encrypt=true;disableStatementPooling=true;databaseName=OctopusSrvConexao;columnEncryptionSetting=Disabled;applicationName=Microsoft JDBC Driver
for SQL Server;applicationIntent=readwrite; (Microsoft SQL Server 16.0)
Successfully validated 5 migrations (execution time 00:00.094s)
Current version of schema [dbo]: 0
Migrating schema [dbo] to version "1" - create crm tables"
Successfully applied 1 migration to schema [dbo], now at version v1 (execution time 00:00.122s)

```

<https://jenkins.banpara.com/job/sandbox/job/Thiago-Oliveira/job/poc-flyway/3/console>

Painel de controle > Sandbox > Thiago Oliveira > poc-flyway > #5

See <https://jenkins.io/redirect/groovy-string-interpolation> for details.

```

+ flyway '-url=jdbc:sqlserver://DESEW-SRVSQL.desenv.com:53679;instanceName=SQLCLUST06;databaseName=OctopusSrvConexao;encrypt=true;trustServerCertificate=true' -user=gen_octopus -password=**** -
locations=filesystem:/home/jenkins/agent/workspace/sandbox/Thiago-Oliveira/poc-flyway/flyway -undo
Picked up JAVA_TOOL_OPTIONS: -Xms1g -Xmx4g -Dhttps.proxyHost=10.0.1.186 -Dhttps.proxyPort=8080 -Dhttps.proxyUser=serviceunix -Dhttps.proxyPassword=Ab123456 -
Dhttp.nonProxyHosts="*.banpara.com|*.homologacao.com|*.desenv.com|10.*.*.*|192.168.*.*|srpserverbi"
Flyway Enterprise Edition 12.0.2-rc1522 by Redgate
WARNING: You are using a limited Flyway trial license, valid until 2026-03-23. In 27 days you must either upgrade to a full Enterprise license or downgrade to Community.

See release notes here: https://rd.gt/4160bNI
Database: jdbc:sqlserver://DESEW-
SRVSQL.desenv.com:SQLCLUST06:53679;connectRetryInterval=10;connectRetryCount=1;maxResultBuffer=-1;sendTemporalDataTypesAsStringForBulkCopy=true;delayLoadingObs=true;useMtlOnly=false;cacheBulkCopyMetadata=false;bulk
CopyForBatchInsertAllowEncryptedValueModifications=false;bulkCopyForBatchInsertTableLock=false;bulkCopyForBatchInsertKeepNulls=false;bulkCopyForBatchInsertKeepIdentity=false;bulkCopyForBatchInsertFireTriggers=false;
bulkCopyForBatchInsertCheckConstraints=false;bulkCopyForBatchInsertBatchSize=0;useBulkCopyForBatchInsert=false;cancelQueryTimeout=-1;sslProtocol=TLS;calBigDecimalPrecision=false;useDefaultJaaSConfig=false;jaasConfi
gurationName=SQLJDBCDriver;statementPoolingCacheSize=0;serverPreparedStatementDiscardThreshold=10;enablePrepareOnFirstPreparedStatementCall=false;fips=false;socketTimeout=0;authentication=NotSpecified;authentication
Scheme=NativeAuthentication;openStates=false;dateTimeParameterType=dateTime;sendTimeAsDateTime=true;replication=false;trustStoreType=JKS;trustServerCertificate=true;transparentNetworkIPResolution=true;IPAddressPre
ference=IPv4First;serverNameAsACE=false;sendStringParametersAsUnicode=true;selectMethod=direct;responseBuffering=adaptive;queryTimeout=-1;packetSize=8000;multiSubnetFailover=false;loginTimeout=30;lockTimeout=-1;last
UpdateCount=true;useDefaultTSSCredential=false;prepareMethod=prepare;encrypt=true;disableStatementPooling=true;databaseName=OctopusSrvConexao;columnEncryptionSetting=Disabled;applicationName=Microsoft JDBC Driver
for SQL Server;applicationIntent=readwrite; (Microsoft SQL Server 16.0)
Current version of schema [dbo]: 1
Undoing migration of schema [dbo] to version "1" - create crm tables"
Successfully undid 1 migration to schema [dbo], now at version v0 (execution time 00:00.490s)

```

<https://jenkins.banpara.com/job/sandbox/job/Thiago-Oliveira/job/poc-flyway/5/console>

- **Observações:** Foi utilizado diretamente os comandos do flyway, sem uso de extensão, no pipeline foi configurada uma licença Trial Enterprise para os testes. Foi possível replicar o mesmo comportamento das execuções manuais. O rollback do flyway não permite especificar um número de versão, então o número de versão é ignorado, ele sempre desfaz a última mudança existente na tabela de changelog.
- **Ferramenta: Liquibase Resultado:**

Para configurar o liquibase no **Dockerfile** do agente **banpara-base** o seguinte trecho.

```

RUN wget --no-check-certificate --user nexus-reader --password nexus-reader
https://nexus.banpara.com/repository/raw-release/FlywayDesktopLinuxX64_9.1.0.0.tar.gz -
P /tmp && \
    tar -xzf /tmp/FlywayDesktopLinuxX64_9.1.0.0.tar.gz -C /opt ./flyway/ && \
    rm -f /tmp/FlywayDesktopLinuxX64_9.1.0.0.tar.gz && \
    chown -R jenkins:jenkins /opt/flyway && \
    ln -s /opt/flyway/flyway /usr/local/bin

```

O pipeline do liquibase é o seguinte:

```

@Library('banpara-esteiras') _

pipeline {
    agent {
        label 'banpara-base'
    }

    options {
        buildDiscarder(logRotator(numToKeepStr: '10'))
        ansiColor('xterm')
    }

    parameters {
        string(
            name: 'TAG_VERSION',
            defaultValue: '1.0',
            description: 'Versão da tag para deploy (ex: 1.0, 2.0)'
        )
        booleanParam(
            name: 'ROLLBACK',
            defaultValue: false,
            description: 'Marcar para executar rollback para a tag especificada'
        )
    }

    environment {
        MSSQL_DRIVER_PATH = '/opt/flyway/drivers/mssql-jdbc-12.10.2.jre11.jar'
        LIQUIBASE_BIN = 'liquibase'
        DB_HOST = 'DESENV-SRVSQL.desenv.com'
        DB_PORT = '53679'
        DB_INSTANCE = 'SQLCLUST06'
        DB_NAME = 'OctopusSrvConexao'
        DB_ENCRYPT = 'true'
        DB_TRUST_SERVER_CERT = 'true'
    }

    stages {
        stage('Checkout') {
            steps {
                checkout([
                    $class: 'GitSCM',
                    branches: [[name: 'principal']],
                    extensions: [],
                    gitTool: 'Default',
                    userRemoteConfigs: [[
                        credentialsId: 'ldap.gen_jenkins.credenciais',
                        url:
'https://gitlab.banpara.com/tfoliveira/octopussrvconexao.git'

```

```

        ]]
    })
}

stage('Executar Liquibase') {
    steps {
        script {
            withCredentials([usernamePassword(
                credentialsId: 'desensrvsqsclust06.gen_octopus.credenciais',
                usernameVariable: 'DB_USER',
                passwordVariable: 'DB_PASSWORD'
            )]) {
                dir('liquibase') {
                    def commonFlags = "--defaultsFile=liquibase.properties " +
                        "--
url=\"jdbc:sqlserver://${DB_HOST}:${DB_PORT};instanceName=${DB_INSTANCE};databaseName=${DB_NAME};encrypt=${DB_ENCRYPT};trustServerCertificate=${DB_TRUST_SERVER_CERT}\" " +
                        "--username=\"${DB_USER}\" " +
                        "--password=\"${DB_PASSWORD}\" " +
                        "--classpath=\"${MSSQL_DRIVER_PATH}\"

                    if (params.ROLLBACK) {
                        sh """
                            ${LIQUIBASE_BIN} ${commonFlags} rollback --
tag=${params.TAG_VERSION}
                        """
                    } else {
                        sh """
                            ${LIQUIBASE_BIN} ${commonFlags} updateToTag
${params.TAG_VERSION}
                        """
                    }
                }
            }
        }
    }
}

post {
    always {
        cleanWs()
    }
}
}

```

O pipeline baixa o repositório e executa o comando do liquibase.

O pipeline tem como parâmetro o número de versão e um indicador de rollback.

Pipeline poc-liquibase

Esta construção requer parâmetros:

TAG_VERSION

Versão da tag para deploy (ex: 1.0, 2.0)

1.0

☐ ROLLBACK

Marcar para executar rollback para a tag especificada

▶ Construir

Cancel

Segue abaixo alguns logs de execução.

Painel de controle > Sandbox > Thiago Oliveira > poc-liquibase > #4

+ liquibase --defaultsFile=liquibase.properties --url=jdbc:sqlserver://DESENV-SRVSQLESENV.com:53679;instanceName=SQLCLUST06;databaseName=OctopusSvcConexao;encrypt=true;trustServerCertificate=true' --username=gen_octopus --password=**** --classpath=/opt/flyway/drivers/mssql-jdbc-12.10.2.jre11.jar updateToTag 1.0
Picked up JAVA_TOOL_OPTIONS: -Xms1g -Xmx4g -Dhttps.proxyHost=10.0.1.186 -Dhttps.proxyPort=8080 -Dhttps.proxyUser=serviceunix -Dhttps.proxyPassword=Ab123456 -Dhttp.nonProxyHosts="*.banpara.com|*.homologacao.com|*.desenv.com|10.*|.*|192.168.*|*|srvpowerbi"

Painel de controle > Sandbox > Thiago Oliveira > poc-liquibase > #4

Starting Liquibase at 12:26:33 using Java 17.0.17 (version 5.0.1 #9400 built at 2025-10-03 17:37+0000)
Liquibase Version: 5.0.1
[2026-02-24 12:26:33] INFO [liquibase.ui] Liquibase Version: 5.0.1

[2026-02-24 12:26:33] INFO [liquibase.ui]
[2026-02-24 12:26:33] INFO [liquibase.integration] Starting command execution.
[2026-02-24 12:26:33] WARNING [liquibase.configuration] Potentially ignored key(s) in file exists at path liquibase.properties
- 'liquibase.hub.mode'
WARNING:

Liquibase detected the following invalid Community LIQUIBASE_* environment variables:

- LIQUIBASE_BIN

Find the list of valid environment variables at <https://docs.liquibase.com/environment-variables>

[2026-02-24 12:26:33] INFO [liquibase.ui] WARNING:

Liquibase detected the following invalid Community LIQUIBASE_* environment variables:

- LIQUIBASE_BIN

Find the list of valid environment variables at <https://docs.liquibase.com/environment-variables>

[2026-02-24 12:26:33] WARNING [liquibase.configuration]

Liquibase detected the following invalid Community LIQUIBASE_* environment variables:

- LIQUIBASE_BIN

Find the list of valid environment variables at <https://docs.liquibase.com/environment-variables>

[2026-02-24 12:26:34] INFO [liquibase.lockservice] Successfully acquired change log lock
[2026-02-24 12:26:35] INFO [liquibase.changelog] Reading from DATABASECHANGELOG
[2026-02-24 12:26:35] INFO [liquibase.command] Using deploymentId: 1935991724
Running Changeset: poc1/changelog-1.0.yaml::1-create-tbCliente::codex
[2026-02-24 12:26:35] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::1-create-tbCliente::codex
[2026-02-24 12:26:35] INFO [liquibase.changelog] Custom SQL executed
[2026-02-24 12:26:35] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::1-create-tbCliente::codex ran successfully in 48ms
Running Changeset: poc1/changelog-1.0.yaml::2-create-tbProduto::codex
[2026-02-24 12:26:35] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::2-create-tbProduto::codex
[2026-02-24 12:26:35] INFO [liquibase.changelog] Custom SQL executed

```
Painel de controle > Sandbox > Thiago Oliveira > poc-liquibase > #4

[2026-02-24 12:26:35] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::2-create-tbProduto::codex ran successfully in 53ms
Running Changeset: poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex
[2026-02-24 12:26:36] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex
[2026-02-24 12:26:36] INFO [liquibase.changelog] Custom SQL executed
[2026-02-24 12:26:36] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex ran successfully in 32ms
Running Changeset: poc1/changelog-1.0.yaml::4-create-tbContato::codex
[2026-02-24 12:26:36] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::4-create-tbContato::codex
[2026-02-24 12:26:36] INFO [liquibase.changelog] Custom SQL executed
[2026-02-24 12:26:36] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::4-create-tbContato::codex ran successfully in 15ms
Running Changeset: poc1/changelog-1.0.yaml::5-create-tbAtividade::codex
[2026-02-24 12:26:36] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::5-create-tbAtividade::codex
[2026-02-24 12:26:36] INFO [liquibase.changelog] Custom SQL executed
[2026-02-24 12:26:36] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::5-create-tbAtividade::codex ran successfully in 23ms
Running Changeset: poc1/changelog-1.0.yaml::6-tag-release-1-0::codex
[2026-02-24 12:26:36] INFO [liquibase.ui] Running Changeset: poc1/changelog-1.0.yaml::6-tag-release-1-0::codex
[2026-02-24 12:26:36] INFO [liquibase.changelog] Tag '1.0' applied to database
[2026-02-24 12:26:36] INFO [liquibase.changelog] ChangeSet poc1/changelog-1.0.yaml::6-tag-release-1-0::codex ran successfully in 2ms

UPDATE SUMMARY
Run: 6
Previously run: 0
Filtered out: 0
-----
Total change sets: 6

[2026-02-24 12:26:36] INFO [liquibase.util] UPDATE SUMMARY
[2026-02-24 12:26:36] INFO [liquibase.util] Run: 6
[2026-02-24 12:26:36] INFO [liquibase.util] Previously run: 0
[2026-02-24 12:26:36] INFO [liquibase.util] Filtered out: 0
[2026-02-24 12:26:36] INFO [liquibase.util] -----
[2026-02-24 12:26:36] INFO [liquibase.util] Total change sets: 6
[2026-02-24 12:26:36] INFO [liquibase.util] Update summary generated
[2026-02-24 12:26:36] INFO [liquibase.command] Update command completed successfully.
Liquibase: Update-to-tag has been successful. Rows affected: 0
[2026-02-24 12:26:36] INFO [liquibase.ui] Liquibase: Update-to-tag has been successful. Rows affected: 0
[2026-02-24 12:26:36] INFO [liquibase.lockservice] Successfully released change log lock
[2026-02-24 12:26:36] INFO [liquibase.lockservice] Successfully released change log lock
[2026-02-24 12:26:36] INFO [liquibase.command] Command execution complete
Liquibase command 'updateToTag' was executed successfully.
[2026-02-24 12:26:37] INFO [liquibase.ui] Liquibase command 'updateToTag' was executed successfully.
```

<https://jenkins.banpara.com/job/sandbox/job/Thiago-Oliveira/job/poc-liquibase/4/console>

```
Painel de controle > Sandbox > Thiago Oliveira > poc-liquibase > #7

+ liquibase --defaultsFile=liquibase.properties --url=jdbc:sqlserver://DESENV-SRVSQL.desenv.com:53679;instanceName=SQLCLUST06;databaseName=OctopusSrvConexao;encrypt=true;trustServerCertificate=true' --
username=gen_octopus -password=**** --classpath=/opt/flyway/drivers/mssql-jdbc-12.10.2.jre11.jar rollback --tag=0.0
Picked up JAVA_TOOL_OPTIONS: -Xms1g -Xmx4g -Dhttps.proxyHost=10.0.1.186 -Dhttps.proxyPort=8080 -Dhttps.proxyUser=servicelinux -Dhttps.proxyPassword=Ab123456 -
Dhttps.nonproxyHosts="*.banpara.com|*.banparagcp.com|*.desenv.com|10.0.1.1|192.168.1.1|srvgwecb1"

Starting Liquibase at 13:23:19 using Java 17.0.17 (version 5.0.1 #9400 built at 2025-10-03 17:37:0000)
Liquibase Version: 5.0.1
[2026-02-24 13:23:19] INFO [liquibase.ui] Liquibase Version: 5.0.1

[2026-02-24 13:23:19] INFO [liquibase.ui]
[2026-02-24 13:23:19] INFO [liquibase.integration] Starting command execution.
[2026-02-24 13:23:19] WARNING [liquibase.configuration] Potentially ignored key(s) in file exists at path liquibase.properties
- 'liquibase.hub.mode'
WARNING:

Liquibase detected the following invalid Community LIQUIBASE_* environment variables:

- LIQUIBASE_BIN

Find the list of valid environment variables at https://docs.liquibase.com/environment-variables

[2026-02-24 13:23:19] INFO [liquibase.ui] WARNING:

Liquibase detected the following invalid Community LIQUIBASE_* environment variables:

- LIQUIBASE_BIN

Find the list of valid environment variables at https://docs.liquibase.com/environment-variables

[2026-02-24 13:23:19] WARNING [liquibase.configuration]

Liquibase detected the following invalid Community LIQUIBASE_* environment variables:

- LIQUIBASE_BIN

Find the list of valid environment variables at https://docs.liquibase.com/environment-variables

[2026-02-24 13:23:22] INFO [liquibase.changelog] Reading from DATABASECHANGELOG
[2026-02-24 13:23:22] INFO [liquibase.lockservice] Successfully acquired change log lock
[2026-02-24 13:23:22] INFO [liquibase.changelog] Reading from DATABASECHANGELOG
Rolling Back Changeset: poc1/changelog-1.0.yaml::8-tag-release-2-0::codex
[2026-02-24 13:23:22] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::8-tag-release-2-0::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::7-change-crm-tables::codex
[2026-02-24 13:23:22] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::7-change-crm-tables::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::6-tag-release-1-0::codex
[2026-02-24 13:23:22] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::6-tag-release-1-0::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::5-create-tbAtividade::codex
```

```
[2026-02-24 13:23:22] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::5-create-tbAtividade::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::4-create-tbContato::codex
[2026-02-24 13:23:23] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::4-create-tbContato::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex
[2026-02-24 13:23:23] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::3-create-tbOportunidade::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::2-create-tbProduto::codex
[2026-02-24 13:23:23] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::2-create-tbProduto::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::1-create-tbCliente::codex
[2026-02-24 13:23:23] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::1-create-tbCliente::codex
Rolling Back Changeset: poc1/changelog-1.0.yaml::0-tag-release-0-0::codex
[2026-02-24 13:23:23] INFO [liquibase.ui] Rolling Back Changeset: poc1/changelog-1.0.yaml::0-tag-release-0-0::codex
[2026-02-24 13:23:23] INFO [liquibase.command] Rollback command completed successfully.
[2026-02-24 13:23:23] INFO [liquibase.lockservice] Successfully released change log lock
[2026-02-24 13:23:23] INFO [liquibase.command] Command execution complete
Liquibase command 'rollback' was executed successfully.
[2026-02-24 13:23:24] INFO [liquibase.ui] Liquibase command 'rollback' was executed successfully.
```

<https://jenkins.banpara.com/job/sandbox/job/Thiago-Oliveira/job/poc-liquibase/7/console>

- **Observações:** O pipeline conseguiu replicar sem problema todos processos executados manualmente.
- **Ferramenta: SSDT Resultado:**

O agente usado para o SSDT foi o `srvjenkinsagent.banpara.com` com a `label` `deploy-windows`, ele já possuía o SSDT instalado, foi instalada apenas a tool do `sqlpackage` com o comando `dotnet tool install -g microsoft.sqlpackage`.

Foi criado o pipeline de build que baixa do repositório o código, constrói e publica no nexus o artefato construído. A url do nexus para visualizar os artefatos é a seguinte

<https://nexus.banpara.com/#browse/browse:maven-releases:Octopus%2FOctopusSrvConexao>.

Segue o pipeline.

```
@Library('banpara-esteiras') _
```

```
pipeline {
    agent {
        label 'deploy-windows'
    }

    options {
        buildDiscarder(logRotator(numToKeepStr: '10'))
        ansiColor('xterm')
    }

    environment {
        GROUP_ID = 'Octopus'
        ARTIFACT_ID = 'OctopusSrvConexao'
        ARTIFACT_TYPE = 'zip'
        SQLPROJ_PATH = 'OctopusSrvConexao/OctopusSrvConexao.sqlproj'
        DIR_ALVO = 'OctopusSrvConexao\\bin\\Output'
    }

    stages {
        stage('Checkout') {
            steps {
                checkout([
                    $class: 'GitSCM',
                    branches: [[name: 'principal']],
                    extensions: [],
                    gitTool: 'Default',
                    userRemoteConfigs: [[
                        credentialsId: 'ldap.gen_jenkins_credenciais',
                        url:
                            'https://gitlab.banpara.com/tfoliveira/octopussrvconexao.git'
                    ]]
                ])
            }
        }
    }
}
```

```

    })
  }
}

stage('Configurar Versão') {
  steps {
    script {
      // Extrai versão do arquivo .sqlproj (substituindo
      dotnetutils.obterNumeroVersao)
      def sqlprojContent = readFile "${SQLPROJ_PATH}"
      def versionStart = sqlprojContent.indexOf("<Version>")
      def versionEnd = sqlprojContent.indexOf("</Version>")
      env.VERSAO = sqlprojContent.substring(versionStart +
      "<Version>".length(), versionEnd).trim()

      // Define nome do artefato
      env.NOME_ARTEFATO = "${GROUP_ID}-${ARTIFACT_ID}-
      v${VERSAO}.${ARTIFACT_TYPE}"

      echo "Versão extraída: ${VERSAO}"
      echo "Nome do artefato: ${NOME_ARTEFATO}"
    }
  }
}

stage('Build') {
  steps {
    script {
      bat """
          msbuild "${SQLPROJ_PATH}" /p:Configuration=Release
/p:Platform="Any CPU"
          """

      // Renomeia o arquivo dacpac usando PowerShell
      powershell """
          Rename-Item -Path
"${WORKSPACE}\\${DIR_ALVO}\\OctopusSrvConexao.dacpac" -NewName "OctopusSrvConexao-
${env:VERSAO}.dacpac"
          """
    }
  }
}

stage('Empacotamento') {
  steps {
    script {
      // Compacta o conteúdo da pasta de output
      zip zipFile: "${NOME_ARTEFATO}", dir: "${WORKSPACE}\\${DIR_ALVO}"
    }
  }
}

stage('Entrega Nexus') {
  steps {
    script {
      // Envia para o Nexus
    }
  }
}

```



```

        esteiraUtils.enviarParaNexus("${NEXUS_MAVEN_RELEASE}")
    }
}

}

}

}

post {
    always {
        cleanWs()
    }
}
}

```

O pipeline foi executado 3 vezes, onde a versão 1.0.0 corresponde ao estado atual do banco, 2.0.0 a versão com os scripts de criação das tabelas e a versão 3.0.0 com alterações de campos. Segue abaixo um log de construção:

```

Panel de controle > Sandbox > Thiago Oliveira > poc-ssdt-build > #18

D:\Jenkins\workspace\sandbox\Thiago-Oliveira\poc-ssdt-build>msbuild "OctopusSrvConexao\OctopusSrvConexao.sqlproj" /p:configuration=Release /p:Platform="Any CPU"
MSBuild version 17.14.40367442088 for .NET Framework
Build started 24/02/2026 12:32:00.

Project "D:\Jenkins\workspace\sandbox\Thiago-Oliveira\poc-ssdt-build\OctopusSrvConexao\OctopusSrvConexao.sqlproj" on node 1 (default targets).
PrepareForBuild:
  Creating directory ".\bin\Output\".
  Creating directory "obj\Any CPU\Release\".
GenerateSqlTargetFrameworkMoniker:
  Skipping target "GenerateSqlTargetFrameworkMoniker" because all output files are up-to-date with respect to the input files.
Compile:
  C:\Program Files\Microsoft Visual Studio\2022\Community\MSBuild\Current\Bin\Roslyn\csc.exe /noconfig /nowarn:1701,1702 /fullpaths /nostdlib+ /highentropyva+ /reference:"C:\Program Files (x86)\Reference Assemblies\Microsoft\Framework\.NETFramework\v4.7.1\mscorlib.dll" /out:"obj\Any CPU\Release\OctopusSrvConexao.dll" /subsystemversion:6.00 /target:library /utf8output /langversion:7.3
  "C:\Users\jgen_ad_jenkins\AppData\Local\Temp\NETFramework,Version=v4.7.1\SqlClrAttributes.cs"
  CompilerServer: server - server processed compilation - OctopusSrvConexao (net472)
SqlBuild:
  Creating a model to represent the project...
  Loading project references...
  Loading project files...
  Building the project model and resolving object interdependencies...
  Validating the project model...
  Writing model to D:\Jenkins\workspace\sandbox\Thiago-Oliveira\poc-ssdt-build\OctopusSrvConexao\obj\Any CPU\Release\Model.xml...
CopyFilesToOutputDirectory:
  Copying file from "D:\Jenkins\workspace\sandbox\Thiago-Oliveira\poc-ssdt-build\OctopusSrvConexao\obj\Any CPU\Release\OctopusSrvConexao.dll" to "D:\Jenkins\workspace\sandbox\Thiago-Oliveira\poc-ssdt-build\OctopusSrvConexao\bin\Output\OctopusSrvConexao.dll".
  OctopusSrvConexao -> D:\Jenkins\workspace\sandbox\Thiago-Oliveira\poc-ssdt-build\OctopusSrvConexao\bin\Output\OctopusSrvConexao.dll
SqlPrepareForRun:
  OctopusSrvConexao -> D:\Jenkins\workspace\sandbox\Thiago-Oliveira\poc-ssdt-build\OctopusSrvConexao\bin\Output\OctopusSrvConexao.dacpac
Done Building Project "D:\Jenkins\workspace\sandbox\Thiago-Oliveira\poc-ssdt-build\OctopusSrvConexao\OctopusSrvConexao.sqlproj" (default targets).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:12.17

```

<https://jenkins.banpara.com/job/sandbox/job/Thiago-Oliveira/job/poc-ssdt-build/18/console>

Para implantação foi criado um outro pipeline com parâmetro de versão que baixa o artefato construído e executa a implantação com sqlpackage. Segue o pipeline.

@Library('banpara-esteiras') _

```

pipeline {
    agent {
        label 'deploy-windows'
    }

    options {
        buildDiscarder(logRotator(numToKeepStr: '10'))
        ansiColor('xterm')
    }

    parameters {
        string(
            name: 'NUMERO_VERSAO',
            defaultValue: '1.0.0',
            description: 'Versão do artefato para deploy (ex: 1.0.0, 2.0.0)'
        )
    }
}

```

```

environment {
    GROUP_ID = 'Octopus'
    ARTIFACT_ID = 'OctopusSrvConexao'
    ARTIFACT_TYPE = 'zip'
    DIR_ALVO = 'bin\\Output'
    DACPAC_NAME = "OctopusSrvConexao-${params.NUMERO_VERSAO}.dacpac"
    SQLPACKAGE_SOURCE = "${WORKSPACE}\\${DIR_ALVO}\\${DACPAC_NAME}"
    TARGET_SERVER = 'DESENV-SRVSQl.desenv.com,53679\\SQLCLUST06'
    TARGET_DB = 'OctopusSrvConexao'
    TARGET_TRUST_SERVER_CERT = 'true'
}

stages {
    stage('Download Artefato') {
        steps {
            script {
                // Baixa o artefato do Nexus
                env.NOME_ARTEFATO =
esteiraUtils.downloadArtefatoNexus(params.NUMERO_VERSAO)

                // Extrai o arquivo ZIP
                powershell """
                    $zipFile = "${WORKSPACE}\\${env.NOME_ARTEFATO}"
                    $extractPath = "${WORKSPACE}\\${DIR_ALVO}"

                    if (-not (Test-Path $extractPath)) {
                        New-Item -ItemType Directory -Path $extractPath -Force
                    }

                    Expand-Archive -Path $zipFile -DestinationPath $extractPath -
Force

                    # Verifica se o arquivo foi extraído corretamente
                    if (-not (Test-Path "${SQLPACKAGE_SOURCE}")) {
                        Write-Host "Arquivo ${DACPAC_NAME} não encontrado após
extração!"

                        exit 1
                    }
                """
            }
        }
    }

    stage('Publicar no Banco') {
        steps {
            script {
                withCredentials([usernamePassword(
                    credentialsId: 'desensrvsqlsqlclust06.gen_octopus.credenciais',
                    usernameVariable: 'TARGET_USER',
                    passwordVariable: 'TARGET_PASSWORD'
                )]) {
                    powershell """
                        sqlpackage.exe /a:Publish `
                        /sf:"${SQLPACKAGE_SOURCE}" `
                        /tsn:"${TARGET_SERVER}" `
                        /tdn:"${TARGET_DB}" `

```

```

        /tu:"${env:TARGET_USER}" `
        /tp:"${env:TARGET_PASSWORD}" `

/TargetTrustServerCertificate:${TARGET_TRUST_SERVER_CERT} `
    /p:DropObjectsNotInSource=True `
    /p:BlockOnPossibleDataLoss=False `

/p:ExcludeObjectTypes="Users;Logins;RoleMembership;Permissions" `
    /p:IgnoreKeywordCasing=True `
    /p:IgnoreWhitespace=True `
    /p:IgnoreWithNocheckOnCheckConstraints=True `
    /p:IgnoreQuotedIdentifiers=True
    """
    }
    }
    }
    }

post {
    always {
        cleanWs()
    }
}
}

```

<https://jenkins.banpara.com/job/sandbox/job/Thiago-Oliveira/job/poc-ssdt-deploy/9/console>

7. ANÁLISE COMPARATIVA

Critério	Flyway	Liquibase	SSDT	Observações
----------	--------	-----------	------	-------------

Tamanho do banco de dados	Não impacta	Não impacta	O projeto vai ficar grande, precisa de todos objetos	
Suporte a rollback	Somente nas versões pagas, possível contornar considerando rollback como uma versão.	Suporta	Suporta	
CI/CD	Suporta	Suporta	Suporta	
Adaptação de scripts	Segue processo similar ao atual. Os scripts precisam apenas seguir nomenclatura própria indicando número de versão e descrição.	A alteração e rollback ficam no mesmo arquivo em um formato específico, permite manter uso de scripts SQL, requer arquivo de configuração.	Ao invés de criar scripts de mudança e rollback, é necessário alterar diretamente os objetos. As mudanças são feitas comparando o pacote com o estado atual do banco.	
Rastreabilidade	Possível através do Git	Possível através do Git	Possível através do Git	
Histórico de mudanças	Armazena em tabela própria	Armazena em tabela própria, rollback remove registros do histórico	Possível através da esteira CD, requer implementação própria	
Impacto em Sistemas com objetos criados em tempo de execução (ex processamento)	Nenhum	Nenhum	Objetos fora do projeto serão removidos durante implantação.	

8. RISCOS E DESAFIOS IDENTIFICADOS

Esta seção consolida os riscos mapeados durante a PoC, organizados por ferramenta avaliada e por dimensão de risco, incorporando também os riscos do contexto real de implantação do projeto Octopus em produção.

Flyway

Riscos Técnicos:

- **Recurso de rollback restrito à versão paga** — o undo está disponível apenas nas edições Teams/Enterprise (Redgate), tornando a reversão manual necessária na versão Community, com risco de inconsistência entre scripts de migração e estado real do banco
- **Dependência do script de rollback** — o arquivo U{versão}__desc.sql precisa ser escrito manualmente e de forma inversa ao V{versão}__desc.sql; erros de codificação nesse script podem levar a estados inconsistentes
- **Histórico vinculado à tabela flyway_schema_history** — se a tabela for corrompida ou apagada acidentalmente, o controle de versão é perdido e o baseline precisa ser refeito
- **Compatibilidade com SQL Server** — a URL JDBC com instanceName e parâmetros de criptografia (trustServerCertificate=true) pode gerar problemas em drivers desatualizados ou configurações de rede restritivas

- **Reconstrução de bases existentes** — para que fosse possível replicar o estado atual de um banco de dados precisaria ser extraído todos scripts de objetos existentes.

Riscos Operacionais:

- **Gestão de versões sequenciais** — em equipes maiores, scripts com versões conflitantes (ex: dois desenvolvedores criam V3__ simultaneamente) resultam em falhas de migração sem mecanismo nativo de resolução de conflitos
- **Licenciamento** — o uso da versão Enterprise em produção sem licença adequada pode gerar problemas de conformidade; o custo de licença por servidor/ano representa impacto orçamentário a ser avaliado

Desafios de Implementação:

- Criação e manutenção disciplinada dos scripts de undo para cada migração
- Definição de convenção de versionamento (numeração de versões) que evite colisões em um ambiente com múltiplos desenvolvedores e múltiplos clusters
- Configuração do agente Jenkins com o binário do Flyway disponível via Nexus (processo já validado na PoC)

Plano de Mitigação Sugerido:

- Adotar política de branches no GitLab que impeça conflitos de versionamento (ex: numeração via sequência centralizada ou timestamp)
- Documentar obrigatoriamente o script de undo junto ao script de migração (U obrigatório para todo V)
- Avaliar custo de licença Teams vs. uso da Community com plano de rollback manual documentado

Liquibase

Riscos Técnicos:

- **Complexidade do changelog em YAML/XML** — a estrutura de múltiplos changesets em um único arquivo pode se tornar difícil de manter à medida que o número de objetos cresce; reorganização posterior pode invalidar checksums já registrados nas tabelas DATABASECHANGELOG
- **Invalidação de checksum** — qualquer modificação em um changeset já executado (mesmo cosmética) gera erro de checksum mismatch, exigindo intervenção manual com clearChecksums
- **Suporte ao undo na versão Community limitado** — rollbacks baseados em tag funcionam apenas se o rollback explícito for definido no changeset; operações automáticas de rollback não são suportadas na versão open source
- **Driver JDBC externo** — a necessidade de incluir o driver mssql-jdbc.jar no classpath exige gestão do artefato do driver e atualização coordenada com versões do SQL Server

Riscos Operacionais:

- **Curva de aprendizado** — o formato YAML/XML com estrutura de changesets é menos intuitivo para DBAs acostumados a scripts SQL puros, podendo aumentar a resistência à adoção
- **Múltiplos arquivos de configuração** — o liquibase.properties com credenciais em texto precisa de tratamento adequado de segredos (integração com Conjur) para evitar exposição de credenciais

Desafios de Implementação:

- Definir e padronizar a estrutura dos changelogs (arquivo único vs. múltiplos arquivos por versão ou por objeto)
- Garantir que cada changeset contenha seu rollback definido explicitamente antes de executar em produção
- Integrar a passagem de credenciais via variáveis de ambiente no pipeline Jenkins, sem uso de arquivo .properties com senhas em texto claro

Plano de Mitigação Sugerido:

- Estabelecer template padrão de changeset com seção de rollback obrigatória
- Usar variáveis de ambiente e cofre de senhas (CyberArk Conjur) para injeção de credenciais no pipeline
- Criar política de imutabilidade para changesets já executados: novas alterações sempre em novos changesets

SQL Server Data Tools (SSDT)

Riscos Técnicos:

- **Objeto não presente no projeto = DROP em produção** — o modelo declarativo do SSDT compara o estado do projeto com o estado do banco; qualquer objeto existente no banco mas ausente no projeto pode ser removido se DropObjectsNotInSource=True não for cuidadosamente gerenciado
- **Objetos criados dinamicamente em runtime** — sistemas que criam tabelas ou objetos temporários/dinâmicos em tempo de execução (como sistemas de processamento em lote) podem ter esses objetos removidos inadvertidamente na próxima implantação
- **Projeto precisa estar em sincronia com o banco** — a importação e manutenção de todos os objetos do banco no projeto exige sincronização contínua; desvios de estado entre projeto e banco resultam em implantações com comportamento inesperado
- **Processo de CI obrigatório** — diferente do Flyway e Liquibase, o SSDT exige a etapa de build (geração do .dacpac) antes do deploy; uma pipeline CI com falha bloqueia toda a implantação

Riscos Operacionais:

- **Dependência do ambiente Windows para build** — o uso do msbuild exige agente Jenkins Windows (deploy-windows), limitando a portabilidade da pipeline e criando um ponto único de falha de infraestrutura
- **Tempo de execução em bases grandes** — reconstrução de tabelas (Altering Table → Starting rebuild table) em bases com grande volume de dados pode gerar indisponibilidade temporária durante a implantação
- **Gestão de constraints** — o parâmetro DoNotDropObjectTypes precisa ser configurado adequadamente para evitar remoção acidental de CheckConstraints e DefaultConstraints, adicionando complexidade operacional

Desafios de Implementação:

- Mapeamento completo e manutenção atualizada de todos os objetos do banco dentro do projeto SSDT
- Definição de política clara para o parâmetro DropObjectsNotInSource — deve ser False por padrão em ambientes produtivos, com exceção explícita documentada e aprovada para cada uso de True
- Gestão de versionamento semântico dos artefatos .dacpac no Nexus (já validado com versões 1.0.0, 2.0.0, 3.0.0)

Plano de Mitigação Sugerido:

- Implementar revisão obrigatória do dacpac em homologação antes de toda implantação em produção
- Utilizar desenvolvimento com dotnet8++ para executar em agentes linux
- Criar checklist de validação pré-deploy que inclua revisão dos parâmetros de Drop e DataLoss
- Documentar e manter a lista de objetos criados dinamicamente que devem ser excluídos do escopo do projeto SSDT

Riscos Transversais (independentes da ferramenta)

Estes riscos se aplicam ao processo de implantação como um todo:

Risco	Impacto	Probabilidade	Mitigação
Resistência de DBAs (GEINS) à adoção de nova ferramenta	Médio	Médio	Envolvimento antecipado, treinamento e co-autoria do processo
Atrasos no processo de mudança GEMUL	Médio	Médio	Documentação completa preparada com antecedência
Credenciais mal configuradas no Conjur impedindo execução dos pipelines	Alto	Baixo	Validação de conectividade antes de cada fase de implantação
Conflito entre objetos gerenciados pela ferramenta e objetos criados por outros processos	Alto	Médio	Delimitar claramente o escopo de objetos gerenciados vs. não gerenciados

Risco	Impacto	Probabilidade	Mitigação
Ambiente de desenvolvimento não reproduzindo fielmente o comportamento de produção	Médio	Médio	Ampliar escopo dos testes para ambiente de homologação antes da produção
Scripts de rollback não testados sendo acionados em produção	Alto	Baixo	Testes obrigatórios de rollback em ambiente de desenvolvimento antes de aprovação

Observação: Os riscos acima foram identificados com base nos cenários executados durante a PoC em ambiente de desenvolvimento (agentes Jenkins e estação de trabalho). Riscos adicionais poderão ser identificados durante a execução do plano de implantação em produção, particularmente nas etapas de baseline nos 6 clusters e configuração dos CronJobs incrementais.

9. ANÁLISE DE CUSTOS

Flyway (Community vs Paid editions)

Flyway oferece três edições principais: Community (gratuita), Teams (paga inicial) e Enterprise (personalizada). A Community é ideal para desenvolvedores individuais, enquanto as pagas adicionam recursos avançados como revisão de código e detecção de drift.

Tabela 1 Comparação de recursos das versões do Flyway

Recurso	Community (Grátis)	Teams/Enterprise (Pago)
Migrações SQL nativas	✓	✓
CLI, API e GUI	✓	✓
Cliente Git integrado	✓	✓
Motor de revisão de código	-	✓
Geração de scripts por IA	-	Preview
Detecção de drift e relatórios	-	✓
Comparação de schema/dados	-	✓ (Enterprise)
Suporte a DBs antigos	-	✓
Suporte a Rollback	-	✓

Essas edições pagas são licenciadas por usuário, com Teams a partir de cerca de US\$ 3.000/ano e Enterprise sob consulta.

Preços Aproximados

- **Community:** Gratuita para uso básico e essencial em versionamento de banco.
- **Teams:** US\$ 3.000/ano (entrada paga, por usuário).
- **Enterprise:** A partir de US\$ 10.000/ano, com descontos por volume e suporte avançado.

Para preços exatos no Brasil ou licenças personalizadas (ex.: para Banpará), contate a Redgate via site oficial. Desde 2025, não há mais edição Pro separada.componentsource+1

Liquibase (Open Source vs Pro)

Liquibase oferece edições Open Source (Community, gratuita), Pro (paga) e Secure (Enterprise, com planos escalonados como Starter, Growth, Business e Enterprise). A Community é básica para migrações, enquanto as pagas adicionam governança, rollbacks avançados e limites por aplicações/databases.

Tabela 2 Limites de Bases/Databases

Plano	Aplicações	Tipos de DB	Schemas/DBs
Community (Grátis)	Ilimitado (básico)	Ilimitado	Sem limite rígido, mas funcionalidades limitadas
Starter	Até 5	1	Até 10 DB <i>targets (legacy)</i>
Growth	Até 10	3	Escalável
Business	Até 25	4	Escalável
Enterprise	Ilimitado	Ilimitado	Ilimitado

"Aplicações" contam sistemas gerenciados; DB targets incluem schemas/produtos. Community não tem limites numéricos, mas falta funcionalidades da versão profissional.

Suporte a Undo/Rollback

Sim, todos os planos suportam rollback básico (rollback-to-tag ou count). Pro/Secure adicionam Targeted Rollback (cherry-pick changeset específico) e auto-geração de scripts.

Preços: Pro a partir de ~US\$ 5.000/ano (10 targets); Enterprise sob consulta. Para setups como SQL Server no Banpará, Secure Enterprise é ideal para escala e compliance.

SSDT

O SSDT (SQL Server Data Tools) é gratuito como workload do Visual Studio 2022, sem custo adicional para instalação ou uso. No entanto, o custo real depende da edição do Visual Studio escolhida, especialmente em ambientes empresariais como o Banpará. Para desenvolvedores DevOps, a Community pode não ser viável devido a restrições corporativas. [learn.microsoft+2](#)

Edições Gratuitas

Visual Studio Community é gratuito e suporta SSDT integralmente, incluindo projetos de banco de dados SQL Server.

Em empresas com >250 PCs ou receita >US\$1M (como bancos), o uso é limitado a cenários open-source ou acadêmicos, exigindo licenças pagas.

Ideal para uso individual ou testes, mas não para produção corporativa.

Edições Pagas

Assinaturas anuais incluem VS 2022 + benefícios como Azure credits e suporte.

Tabela 3 Comparação de assinaturas do VS2022

Edição	Preço Inicial Anual (USD)	Preço Renovação (USD/mês)	Preço Aprox. BRL Anual*	Indicado Para
Professional Standard	\$99.99/mês (\$1.200/ano)	\$66.59/mês	R\$5.300 - R\$6.700	Equipes pequenas/médias, CI/CD básico visualstudio.microsoft+1
Enterprise Standard	\$499.92/mês (\$6.000/ano)	\$214.09/mês	R\$26.500+	DevOps avançado, testes enterprise, multi-agentes

*Média conversão USD/BRL ~5.6; volume reduz até 50% via Open Value ou Enterprise Agreement.

Opções Mensais e Alternativas

- Mensal: Professional \$45 USD/mês (~R\$250), Enterprise \$250 USD/mês (~R\$1.400) – flexível, sem compromisso anual.

- Licenças ESD perpétuas (sem subscription): ~R\$2.000 - R\$3.000 para Professional (varejo), mas sem updates/benefícios.
Para Banpará, verificar volume Microsoft para descontos; SSDT não adiciona custo extra.

Considerações para DevOps/DBA

Em transições para AI agents (LangGraph/LangChain), Enterprise oferece IntelliTest e Live Share úteis para SQL + Python.

Custo total: R\$0 se Community viável; senão, R\$5k+/ano por dev.

10. CONCLUSÕES E RECOMENDAÇÕES

A prova de conceito, conduzida entre 18/02/2026 e 18/03/2026, avaliou três ferramentas de versionamento e implantação de banco de dados — Flyway, Liquibase e SQL Server Data Tools (SSDT) — em três cenários práticos: criação inicial de estrutura, alteração de estrutura existente e integração com pipeline CI/CD, todos aplicados sobre o SQL Server do ambiente Banpará (agentes Jenkins e estação de trabalho).

Os testes comprovaram que todas as três ferramentas são tecnicamente viáveis para integração com o ambiente atual (Jenkins + GitLab + SQL Server), com diferenças relevantes em modelo operacional, curva de adoção, suporte a rollback.

Considerando o processo atual o flyway é o que possui menor impacto e poderia ser implementado de forma imediata, o liquibase permite gerar as mudanças também através de SQL mudando apenas que o rollback precisa ficar no mesmo arquivo e possui baixo impacto de mudança. O SSDT por sua vez é a ferramenta mais segura no sentido que o rollback é gerado de forma automática e evita erro humano, facilita a comparação de objetos e teria um processo similar ao de aplicações para execução.

Recomenda-se uma abordagem mista, usar o SSDT de forma geral, mas para projetos com objetos dinâmicos gerados pelos sistemas ou com uma grande quantidade de objetos usar liquibase ou flyway.

O próximo passo é apresentar e avaliar com os envolvidos DBA e áreas de sistemas as ferramentas para decidir qual *roadmap* seguir.