

Complementos de Bases de Dados 2019/2020

Licenciatura em Eng^a. Informática

2^a Fase Relatório Técnico

Turma: SW 2

Horário de Laboratório: 6^a Feira 16:30h

Docente: João Lázaro

Grupo nº 6

nº190221093 Alexandre Coelho

nº190221128 Sérgio Veríssimo

1. Sumário Executivo

Este projeto teve a sua realização no âmbito da unidade curricular de Complementos de Base de Dados. O objetivo deste incide sobre a reestruturação da base de dados de modo a suportar a gestão de forma integrada, todo o processo de vendas da companhia AdventureWorks e AdventureServices. Esta plataforma deve permitir que seja feita a gestão da informação de todas as companhias existentes, conter também um sistema de autenticação através do email e password e recuperação de password.

2. Especificação de Requisitos

Neste capítulo são descritos e estabelecidos os requisitos funcionais e os atores do sistema intrínsecos à plataforma a ser desenvolvida para o grupo Adventure.

2.1 Requisitos funcionais

Os requisitos aqui apresentados tiveram por fundamento os objetivos traçados ao longo do planeamento do projeto.

ID	Descrição	Implementado (S/N)
RF01	O sistema deverá permitir a existência de utilizadores de acesso à aplicação.	S
RF02	O sistema deverá permitir a autenticação perante a aplicação será feita com recurso à conta de email e password.	S
RF02.1	O sistema deverá permitir que para os novos utilizadores a password do sistema seja alvo de reset na primeira entrada.	S
RF02.2	O sistema deverá permitir sempre que é adicionado um novo utilizador ou este solicita a recuperação de password, o sistema deverá automaticamente gerar uma password e enviar e-mail ao mesmo com essa informação.	S
RF03	O sistema deverá permitir que todas as informações de vendas sejam divididas por companhia e que a informação relativa aos clientes e produtos sejam comuns a todas as companhias.	S
RF04	O sistema deverá permitir que os produtos estejam organizados em Sub-Categorias que por sua vez se agrupam em Categorias mais gerais.	S
RF03	O sistema deverá permitir que todas as informações de vendas sejam divididas por companhia e que a informação relativa aos clientes e produtos sejam comuns a todas as companhias.	S
RF04	O sistema deverá permitir ao utilizador escolher pelo menos 3 questões de recuperação de conta.	S

2ª Fase Relatório Técnico – Complementos de Bases de Dados

RF05	O sistema deverá permitir a gestão de utilizadores.	S
RF04.1	O sistema deverá permitir a edição de utilizadores.	S
RF04.2	O sistema deverá permitir a inserção de utilizadores.	S
RF04.3	O sistema deverá permitir a remoção de utilizadores.	S
RF04.4	O sistema deverá permitir ao utilizador recuperar a sua password, através das perguntas de segurança definidas anteriormente (pelo menos 3).	S
RF05	O sistema deverá permitir a gestão de produtos, categorias, sub-categorias.	S
RF05.1	O sistema deverá permitir a edição de produtos, categorias, sub-categorias.	S
RF05.2	O sistema deverá permitir a inserção de produtos, categorias, sub-categorias.	S
RF05.3	O sistema deverá permitir a remoção de produtos, categorias, sub-categorias.	S
RF05.4	O sistema deverá permitir a associação de produtos a sub-categorias e categorias.	S
RF05.5	O sistema deverá permitir a definição de uma promoção na encomenda.	S
RF05.6	O sistema deverá permitir a alteração das datas de início e fim de uma promoção.	S
RF05.7	O sistema não deverá permitir a atribuição de uma promoção que não esteja ativa.	S
RF05.8	O sistema deverá permitir a alteração do estado dos produtos.	S

2ª Fase Relatório Técnico – Complementos de Bases de Dados

2.2 Atores

Nome	Descrição
Administrador	Cabe ao administrador gerir toda a aplicação.
Utilizador Registado	Este pode consultar todas as sales e customers. Este encontra-se autenticado.
Utilizador Anónimo	Este pode consultar os products. Este não se encontra autenticado.
Gestor de Marketing	Gere a secção de marketing, tendo acesso aos products e as sales.

3. Modelo Relacional (*Modelo de dados*)

Neste tópico é descrito o modelo relacional da base de dados da plataforma de gestão.



Figura 1 - Modelo Relacional

2ª Fase Relatório Técnico – Complementos de Bases de Dados

Legenda:

- Roxo – Sales Schema;
- Azul – Customers Schema;
- Vermelho – Products Schema;
- Laranja – Managements Schema;
- Verde – Accounts Schema.

4. Dimensionamento do Layout

Neste capítulo será descrito e definido o dimensionamento da base dados a ser desenvolvida neste projeto. Assim como os seus filegroups e os discos onde estes filegroups pertencem.

4.1 Filegroups

Filegroup	Tabelas/Ficheiros	RAID/Disco	Detalhes
Primary	Contém o ficheiro com os dados necessários para a inicialização da base de dados (ficheiro .mdf).	Raid 5	É necessário que exista um acesso de leitura rápido ao ficheiro de inicialização da base de dados, pois contém informação crítica e ao mesmo tempo redundância para salvaguardar os dados.
Logs	Armazena o registo de transações e a informação necessária à recuperação da base de dados (ficheiro .ldf).	Raid 1	Estas tabelas / ficheiros necessitam que exista uma grande redundância e velocidade de leitura, pois contém todas as ações realizadas na base de dados.
Products_Filegroup	Armazena os registos que se encontram nas tabelas do schema Products.	Raid 5	Estas tabelas / ficheiros estão frequentemente a ser acedidas por modo de leitura, necessitando de uma grande velocidade de leitura.
Sales_Filegroup	Armazena os registos que se encontram nas tabelas do schema Sales.	Raid 10	Estas tabelas / ficheiros estão frequentemente a ser acedidas por modo de escrita, mas ao mesmo tempo permitir alguma segurança nos dados.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

Customer_Filegroup	Armazena os registos que se encontram nas tabelas do schema Customers.	Raid 5	Estas tabelas / ficheiros estão frequentemente a ser acedidas por modo de leitura, mas ao mesmo tempo precisam de redundância nos dados.
Accounts_Filegroup	Armazena os registos que se encontram nas tabelas do schema Accounts.	Raid 5	Estas tabelas / ficheiros estão frequentemente a ser acedidas por modo de leitura, mas ao mesmo tempo precisam de redundância nos dados.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

4.2 Espaço ocupado por registo de cada tabela

Tabela	Colunas	Tamanho de cada registo (por tabela, em bytes)
[Accounts].[SecurityQuestion]	SecurityQuestionKey int, QuestionText varchar(150)	156 Bytes
[Accounts].[sentEmails]	SentEmailKey int, UserAccountKey int, EmailAddress varchar(100), NewPassword varbinary(128)	218 Bytes
[Accounts].[UserAccount]	UserAccountKey int, Email varchar(100), PasswordHash varbinary(128)	214 Bytes
[Accounts].[UserAccount_SecurityQuestion]	UserAccountSecurityQuestionKey int, SecurityQuestionKey int, UserAccountKey int, AnswerText varchar(100)	114 Bytes
[Accounts].[UserAuthentication]	UserLogsKey int, LoginDate datetime, LogoutDate datetime,	24 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

	UserAccountKey int	
[Customers].[CustomerCommuteDistance]	CustomerCommuteDistanceKey int, CommuteDistanceName varchar(15)	21 Bytes
[Customers].[CustomerCountryRegion]	CustomerCountryRegionKey int, CountryRegionCode varchar(5), CountryRegionName varchar(50)	63 Bytes
[Customers].[CustomerEducation]	CustomerEducationKey int, EducationName varchar(25)	31 Bytes
[Customers].[CustomerOccupation]	CustomerOccupationKey int, OccupationName varchar(25)	31 Bytes
[Customers].[CustomerStateProvince]	CustomerStateProvinceKey int, StateProvinceCode varchar(5), StateProvinceName varchar(50)	63 Bytes
[Customers].[CustomerYearlyIncome]	CustomerYearlyIncomeKey int, YearlyIncome money	12 Bytes
[Managements].[ChangeLog]	UserChangeKey INT, Operation VARCHAR(1), TableName VARCHAR(50), ChangeDateTime DATETIME2	75 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

[Managements].[Error]	ErrorKey int, ErrorName varchar(100)	106 Bytes
[Managements].[ErrorLog]	ErrorLogKey int, ErrorKey int, UserAccountKey int, ErrorDate datetime	12 Bytes
[Managements].[ProductCategoryLog]	ProductCategoryLogKey INT, ProductCategoryKey INT, EnglishProductCategoryName VARCHAR(50), SpanishProductCategoryName VARCHAR(50), FrenchProductCategoryName VARCHAR(50)	164 Bytes
[Managements].[ProductLog]	ProductLog INT, ProductKey INT, ListPrice MONEY, FinishedGoodsFlag BIT, ProductColor INT, SafetyStockLevel INT, StandardCost MONEY, Weight FLOAT, DaysToManufacture TINYINT, ProductLine VARCHAR(1),	91 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

	DealerPrice MONEY, Class VARCHAR(1), Style VARCHAR(1), Status VARCHAR(10), ProductModelKey INT, ProductDescriptionKey INT, ProductNameKey INT, CategorySubCategoryKey INT, ProductMeasureUnitCodeKey INT, SizeSizeRangeKey INT	
[Managements].[ProductSubCategoryLog]	ProductSubCategoryLogKey INT, ProductSubCategoryKey INT, EnglishProductSubCategoryName VARCHAR(50), SpanishProductSubCategoryName VARCHAR(50), FrenchProductSubCategoryName VARCHAR(50)	164 Bytes
[Managements].[UserLog]	UserLog INT, UserAccountKey INT, Email VARCHAR(100), PasswordHash VARBINARY(128)	238 Bytes
[Products].[Category_SubCategory]	CategorySubCategoryKey int, ProductCategoryKey int,	12 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

	ProductSubCategoryKey int	
[Products].[Product]	ProductKey int, ListPrice money, FinishedGoodsFlag bit, ProductColor int, SafetyStockLevel int, StandardCost money, Weight float, DaysToManufacture tinyint, ProductLine varchar(1) DealerPrice money, Class varchar(1), Style varchar(1), Status varchar(10), ProductModelKey int, ProductDescriptionKey int, ProductNameKey int, CategorySubCategoryKey int, ProductMeasureUnitCode int, SizeSizeRangeKey int	87 Bytes
[Products].[ProductCategory]	ProductCategoryKey int,	160 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

	EnglishProductCategoryName varchar(50), SpanishProductCategoryName varchar(50), FrenchProductCategoryName varchar(50)	
[Products].[ProductColor]	ProductColorKey int, ColorName varchar(15)	21 Bytes
[Products].[ProductDescription]	ProductDescriptionKey int, EnglishDescription varchar(300), FrenchDescription varchar(300)	608 Bytes
[Products].[ProductMeasureUnitCode]	ProductMeasureUnitCodeKey int, WeightUnitMeasureCode varchar(2), SizeUnitMeasureCode varchar(2)	12 Bytes
[Products].[ProductModel]	ProductModelKey int, ModelName varchar(100)	106 Bytes
[Products].[ProductName]	ProductNameKey int, EnglishProductName varchar(100), SpanishProductName varchar(100), FrenchProductName varchar(100)	310 Bytes
[Products].[ProductPromotion]	ProductPromotionKey int, StartDate date, EndDate date,	25 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

	Active bit, PriceDiscount money, ProductKey int	
[Products].[ProductSize]	ProductSizeKey INT, Size VARCHAR(5)	11 Bytes
[Products].[ProductSizeRange]	ProductSizeRangeKey INT, SizeRange VARCHAR(15)	21 Bytes
[Products].[ProductSubCategory]	ProductSubCategoryKey int, EnglishProductSubCategoryName varchar(50), SpanishProductSubCategoryName varchar(50), FrenchProductSubCategoryName varchar(50)	160 Bytes
[Products].[Size_SizeRange]	SizeSizeRangeKey INT, ProductSizeKey INT, ProductSizeRangeKey INT	12 Bytes
[Products].[ProductComment]	ProductCommentKey INT , CommentText VARCHAR(150), CommentClassification INT, CommentDate DATE, CustomerEmail VARCHAR(150), ProductKey int	50 Bytes
[Sales].[SalesCompany]	SalesCompanyKey INT, CompanyName VARCHAR(50)	56 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

[Sales].[SalesCurrency]	SalesCurrencyKey INT, CurrencyAlternateKey VARCHAR(50), CurrencyName VARCHAR(50)	108 Bytes
[Sales].[SalesOrderDetail]	SalesOrderDetailKey INT, OrderQuantity INT, UnitPrice MONEY, ProductKey INT, SalesOrderLineNumber TINYINT, SalesOrderHeaderKey INT	25 Bytes
[Sales].[SalesOrderHeader]	SalesOrderHeaderKey INT, SalesOrderNumber VARCHAR(15), OrderDate DATE, DueDate DATE, ShipDate DATE, SalesAmount MONEY, SalesTerritoryKey INT, CustomerKey INT, SalesCompanyKey INT, SalesCurrencyKey INT, SalesStatusKey INT	69 Bytes
[Sales].[SalesTerritory]	SalesTerritoryKey INT,	85 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

	SalesTerritoryRegion VARCHAR(25), SalesTerritoryGroup VARCHAR(25) , SalesTerritoryCountry VARCHAR(25)	
[Sales].[SalesStatus]	SalesStatusKey INT, StatusName VARCHAR(50)	56 Bytes

2ª Fase Relatório Técnico – Complementos de Bases de Dados

4.3 Espaço ocupado pelas tabelas

Os valores abaixo foram obtidos através do procedure **sp_spaceused** e seguiram o seguinte raciocínio:

- Para tabelas com menos de 1000 registo foi assumido o valor de 1000 registos.
- O crescimento é calculado para 5 anos;
- Se número de registos da tabela ocupa X e queremos saber quanto ocupa 1000 registos, então Initial Size = $(X * 1000) / \text{número de registos da tabela}$.
- Para calcular o max size, é feito $(\text{Initial Size} * \text{Percentagem de Crescimento}) ^ 5$.

Tabelas	Rows Size	Filegrowth (%)	Initial Size (KB)	Max Size (KB)
[Accounts].[SecurityQuestion]	0	8	512 (0)	
[Accounts].[sentEmails]	0	20	512 (0)	
[Accounts].[UserAccount]	0	20	512 (0)	
[Accounts].[UserAccount_SecurityQuestion]	0	10	512 (0)	
[Accounts].[UserAuthentication]	0	30	512 (0)	
[Customers].[Customer]	18484	20	4936	$((4936 \times 0,2)^5) + 4936 = 5926$
[Customers].[CustomerCommuteDistance]	5	8	$(1000 * 72) / 5 = 14400$	$((14400 \times 0,08)^5) + 14400 = 15557$
[Customers].[CustomerCountryRegion]	6	8	$(1000 * 72) / 6 = 12000$	$((12000 \times 0,08)^5) + 12000 = 12965$

2ª Fase Relatório Técnico – Complementos de Bases de Dados

[Customers].[CustomerEducation]	5	8	$(1000 \times 72) / 5 = 14400$	$((14400 \times 0,08)^5) + 14400$ = 15557
[Customers].[CustomerOccupation]	5	8	$(1000 \times 72) / 5 = 14400$	$((14400 \times 0,08)^5) + 14400$ = 15557
[Customers].[CustomerStateProvince]	53	8	$(1000 \times 72) / 53 = 1358$	$((1358 \times 0,08)^5) + 1358 =$ 1463
[Customers].[CustomerYearlyIncome]	16	8	$(1000 \times 72) / 16 = 4500$	$((4500 \times 0,08)^5) + 4500 =$ 4865
[Managements].[ChangeLog]	0	40	512 (0)	
[Managements].[Error]	0	8	512 (0)	
[Managements].[ErrorLog]	0	50	512 (0)	
[Managements].[ProductCategoryLog]	0	40	512 (0)	
[Managements].[ProductLog]	0	40	512 (0)	
[Managements].[ProductSubCategoryLog]	0	40	512 (0)	
[Managements].[UserLog]	0	40	512 (0)	

2ª Fase Relatório Técnico – Complementos de Bases de Dados

[Products].[Category_SubCategory]	37	8	$(1000 \times 72) / 37 = 1946$	$((1946 \times 0,08)^5) + 1946 = 2140$
[Products].[Product]	397	25	$(1000 \times 72) / 397 = 181$	$((181 \times 0,25)^5) + 181 = 221$
[Products].[ProductCategory]	4	8	$(1000 \times 72) / 4 = 18000$	$((18000 \times 0,08)^5) + 18000 = 19445$
[Products].[ProductColor]	10	5	$(1000 \times 72) / 10 = 7200$	$((7200 \times 0,05)^5) + 7200 = 7565$
[Products].[ProductDescription]	114	8	$(1000 \times 72) / 114 = 632$	$((632 \times 0,08)^5) + 632 = 687$
[Products].[ProductMeasureUnitCode]	2	2	$(1000 \times 72) / 2 = 36000$	$((36000 \times 0,02)^5) + 36000 = 36725$
[Products].[ProductModel]	119	8	$(1000 \times 72) / 119 = 605$	$((605 \times 0,08)^5) + 605 = 658$
[Products].[ProductName]	295	8	$(1000 \times 72) / 295 = 244$	$((244 \times 0,08)^5) + 244 = 266$
[Products].[ProductPromotion]	0	15	512 (0)	
[Products].[ProductSize]	18	5	$(1000 \times 72) / 18 = 4000$	$((4000 \times 0,05)^5) + 4000 = 4205$
[Products].[ProductSizeRange]	10	5	$(1000 \times 72) / 10 = 7200$	$((7200 \times 0,05)^5) + 7200 = 7565$

2ª Fase Relatório Técnico – Complementos de Bases de Dados

[Products].[ProductSubCategory]	37	8	$(1000 \times 72) / 37 = 1946$	$((1946 \times 0,08)^5) + 1946 = 2140$
[Products].[Size_SizeRange]	18	5	$(1000 \times 72) / 18 = 4000$	$((4000 \times 0,05)^5) + 4000 = 4205$
[Sales].[SalesCompany]	2	0	$(1000 \times 72) / 2 = 36000$	$((36000 \times 0)^5) + 36000 = 36005$
[Sales].[SalesCurrency]	105	5	$(1000 \times 72) / 105 = 686$	$((686 \times 0,05)^5) + 686 = 725$
[Sales].[SalesOrderDetail]	60398	30	2056	$((2056 \times 0,30)^5) + 2056 = 2677$
[Sales].[SalesOrderHeader]	17991	30	1744	$((1744 \times 0,30)^5) + 1744 = 2270$
[Sales].[SalesTerritory]	11	8	$(1000 \times 72) / 11 = 6545$	$((6545 \times 0,08)^5) + 6545 = 7071$
[Sales].[SalesStatus]	3	2	$(1000 \times 72) / 3 = 24000$	$((24000 \times 0,02)^5) + 24000 = 24485$

5. Schemas

Nome	Descrição
Customers	Este schema possui como objetivo acoplar todas as tabelas, stored procedures e funções inerentes aos clientes da plataforma.
Products	Este schema possui como objetivo acoplar todas as tabelas, stored procedures e funções inerentes aos produtos que são comercializados na plataforma.
Sales	Este schema possui como objetivo acoplar todas as tabelas, stored procedures e funções inerentes as vendas realizadas através da plataforma.
Accounts	Este schema possui como objetivo acoplar todas as tabelas, stored procedures e funções inerentes a autenticação e gestão de contas da plataforma.
Managements	Este schema possui como objetivo acoplar todas as tabelas, stored procedures e funções inerentes ao tratamento de erros e backup de dados da plataforma.

6. Views

Nome	Descrição
Sales.SalesPerYear_VW	Esta view permite obter o total monetário de vendas por ano da AdventureNewData.
Sales.SalesPerYearAndCountry_VW	Esta view permite obter o total monetário de vendas por ano por “Sales Territory Country” da AdventureNewData.
Sales.SalesPerYearAndSubcategory_VW	Esta view permite obter o total monetário de vendas por ano por “Product Subcategory” da AdventureNewData.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

Sales.SalesPerYearAndCategory_VW	Esta view permite obter o total monetário de vendas por ano por “Product Category” da AdventureNewData.
Sales.NumberOfClientsPerYearAndSalesTerritoryCountry_VW	Esta view permite obter o número de Clientes por ano por “Sales Territory Country” da AdventureNewData.
Sales.SalesPerYearOld_VW	Esta view permite obter o total monetário de vendas por ano da AdventureOldData.
Sales.SalesPerYearAndCountryOld_VW	Esta view permite obter o total monetário de vendas por ano por “Sales Territory Country” da AdventureOldData.
Sales.SalesPerYearAndSubcategoryOld_VW	Esta view permite obter o total monetário de vendas por ano por “Product Subcategory” da AdventureOldData.
Sales.SalesPerYearAndCategoryOld_VW	Esta view permite obter o total monetário de vendas por ano por “Product Category” da AdventureOldData.
Sales.NumberOfClientsPerYearAndSalesTerritoryCountryOld_VW	Esta view permite obter o número de Clientes por ano por “Sales Territory Country” da AdventureOldData.

7. Functions

Nome	Tipo	Atributos	Descrição
Accounts.hashPassword	Varbinary	@password varchar(100)	Permite gerar um hash da password recebida como atributo.

8. Stored procedures

Nome	Atributos	Descrição
Accounts.spLogout	@Email varchar(100)	Permite terminar sessão com a sua conta.
Accounts.spLogin	@Email varchar(100), @Password varchar(80)	Permite realizar a autenticação do user.
Accounts.spAddQuestion User	@SecurityQuestionKey int, @Answer varchar(100), @UserKey int	Permite adicionar uma pergunta de recuperação ao utilizador.
Accounts.spNewUserAccount	@Email varchar(250)	Permite criar um novo utilizador.
Sales.SalesCreation	@EmailAddress varchar(60), @ProductName varchar(100), @Quantity int, @CurrencyOfPayment varchar(3)	Cria Sales(Vendas).
Sales.AddAProductToOrder	@ProductName varchar(100), @Quantity int, @SalesOrderNumber varchar(15)	Adiciona produtos a Encomenda.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

Sales.spAlterQuantityOnOrder	@ProductName varchar(50), @Quantity int, @SalesOrderNumber varchar(10)	Altera quantidade de Produto na Encomenda.
Sales.spRemoveProductFromOrder	@ProductName varchar(50), @SalesOrderNumber varchar(10)	Remover Produto de Encomenda.
Sales.spAlterOrderState	@OrderKey int, @Status int	Alterar Estado da Encomenda.
[Managements].spGenerator	@Schema Varchar(40), @TableName varchar(100)	Procedure Gerador de outros procedures.
Products.spAssociateProductSubCategory	@ProductName varchar(50), @SubCategoryName varchar(50)	Associar Produto a SubCategoria.
Sales.spAssociatePromotionToSale	@SalesOrderNumber varchar(13), @PromotionKey int	Associar Promoção a Encomenda.
Managements.InsertErrors		Inserir os erros na tabela errors.
Accounts.InsertSecurityQuestions		Inserir as perguntas de segurança na tabela SecurityQuestions.
Accounts.spRecoverPassword	@Email varchar(100), @NewPassword varchar(80), @Answer1 varchar(100), @Answer2 varchar(100), @Answer3 varchar(100)	Recupera a password do utilizador.
[Products].populateComments		Gera vários comentários e popula a tabela ProductComment.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

[Products].newComment	@CommentText varchar(150), @CommentClassification varchar(50), @CommentDate date, @ProductKey int, @CustomerEmail varchar(150)	Cria um novo comentário.
-----------------------	--	--------------------------

9. Triggers

Nome	Tipo	Tabela	Descrição
[Products].ProductLog	AFTER UPDATE, AFTER DELETE, AFTER INSERT	[Products].Product	Guarda o histórico de alterações sobre o produto.
[Products].ProductSubCategoryLog	AFTER UPDATE, AFTER DELETE, AFTER INSERT	[Products].ProductSubCategory	Guarda o histórico de alterações sobre a subcategoria do produto.
[Products].ProductCategoryLog	AFTER UPDATE, AFTER DELETE, AFTER INSERT	[Products].ProductCategory	Guarda o histórico de alterações sobre a categoria do produto.

10. Consultas

10.1 Verificação da conformidade dos dados

Foram criadas várias view com o intuito de verificar a conformidade dos dados do novo modelo. Estas views encontram-se descritas no capítulo 6.

11. Índices

Designação	Tabela	Justificação/Consultas
_dta_index_SalesOrderDetail_11_114099447__K4_2	[Sales].[SalesOrderDetail]	(nonclusteredindex): foi adicionada a chave de pesquisa que especifica uma ordem diferente da ordem sequencial do ficheiro.
_dta_index_SalesOrderDetail_11_114099447__K5	[Sales].[SalesOrderDetail]	(nonclusteredindex): foi adicionada a chave de pesquisa que especifica uma ordem diferente da ordem sequencial do ficheiro.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

_dta_stat_114099447_5_4	[Sales].[SalesOrderDetail]	statistics: foi criado para ajudar o SQL Server Engine a determina a cardinalidade de diferentes valores numa coluna
_dta_index_SalesOrderDetail_11_114099447__K6_2_3	[Sales].[SalesOrderDetail]	(nonclusteredindex): foi adicionada a chave de pesquisa que especifica uma ordem diferente da ordem sequencial do ficheiro.
_dta_index_SalesOrderHeader_11_2133582639__K1_K7_3	[Sales].[SalesOrderHeader]	(nonclusteredindex): foi adicionada a chave de pesquisa que especifica uma ordem diferente da ordem sequencial do ficheiro.
_dta_stat_2133582639_7_1	[Sales].[SalesOrderHeader]	statistics: foi criado para ajudar o SQL Server Engine a determina a cardinalidade de diferentes valores numa coluna
_dta_index_SalesOrderHeader_11_2133582639__K7_3_6	[Sales].[SalesOrderHeader]	(nonclusteredindex): foi adicionada a chave de pesquisa que especifica uma ordem diferente da ordem sequencial do ficheiro.
_dta_index_SalesOrderHeader_11_2133582639__K7_3_5	[Sales].[SalesOrderHeader]	(nonclusteredindex): foi adicionada a chave de pesquisa que especifica uma ordem diferente da ordem sequencial do ficheiro.

12. Otimização e Execução de Consultas

Efetuamos a execução das queries necessárias por ordem do enunciado, e anexamos no ficheiro do projeto os trace files sem a configuração de índices e com a configuração de índices. O nome do ficheiro está representado como por exemplo: Query1-NoIndex, o que representa a execução da query 1 (total de vendas por produto) sem indexação, e sendo que Query1-WithIndex, representa a execução da query 1 com indexação. Os ficheiros referido anteriormente, encontram-se na pasta Tracefiles.

Anexamos aqui no relatório a duração das queries, para facilitar a compreensão ao avaliador de projeto:

Query 1:

Antes: 117ms

Depois: 100ms

Query 2:

Antes: 23ms

Depois: 26ms

2ª Fase Relatório Técnico – Complementos de Bases de Dados

Apesar de ter mais 3ms na execução a seguir à configuração, a indexação pode ser a mais recomendada de momento, mas nem sempre indica que coloca a query mais veloz. Também pode existir externalidades que influenciam o tempo de execução. Refiro aqui também, quanto à segunda query

Query 3:

Antes: 144ms

Depois: 83ms

Query 4:

Antes: 48ms

Depois: 27ms

Query 5:

Antes: 14ms

Depois: 21ms

Repete-se a mesma situação que a query 2.

13. Controlo de Concorrência

Foram criadas várias transações dentro dos procedures que foram pedidos no enunciado. Esta iram salvaguardar a integridade da base de dados, caso ocorra algum erro durante a execução dos procedures.

14. Segurança e Controlo de Acessos

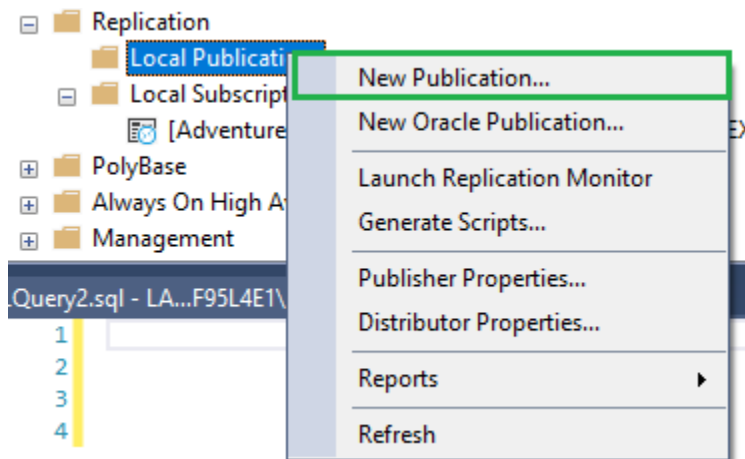
Na definição de utilizadores, roles e schemas, foram definidos os vários utilizadores, roles e permissões a schemas conforme foi pedido no enunciado. Estas criações encontram-se nos ficheiros **13 - Access & Privileges** e **14 - Access & Privileges – Procedures**.

Na encriptação foi utilizado o hashing para a password, pois este não é reversível, mas permite a comparação e validação da password no momento da autenticação.

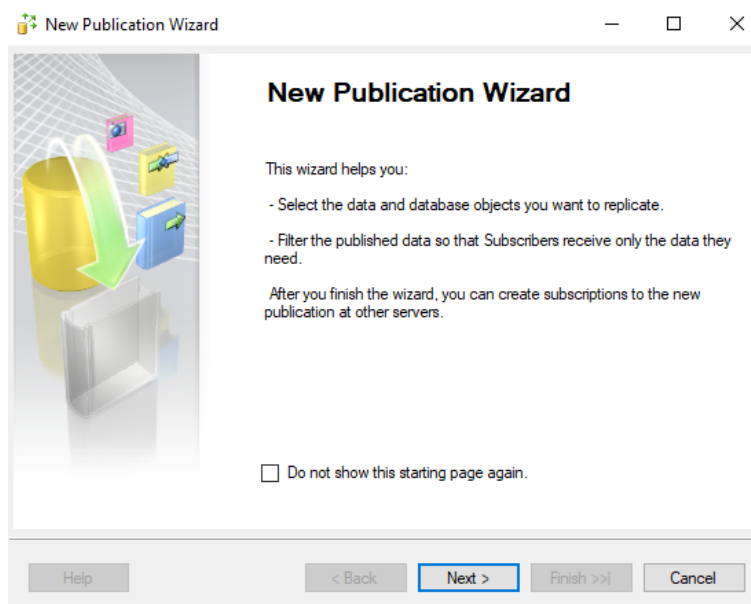
15. Esquema de Replicação

Criação de um publisher

Na replicação, começamos por criar um novo publisher, processo este que se inicia com um clique com o botão direito do rato em “Local Publication” (que se encontra dentro da pasta “Replication”). Posteriormente, seleccionamos a opção “New Publication...”.

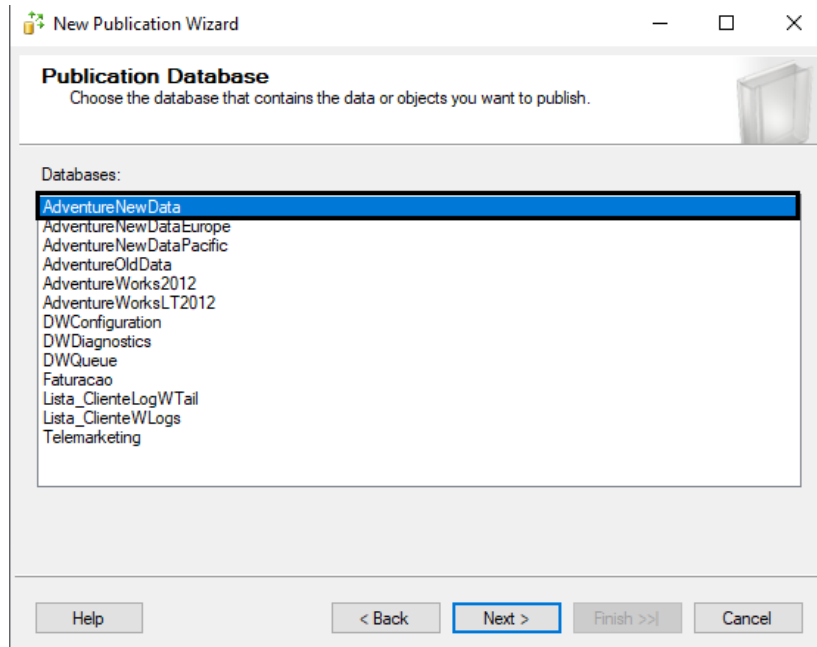


Após o último passo somos encaminhados para o “New Publication Wizard”, onde se inicia a configuração do publisher.

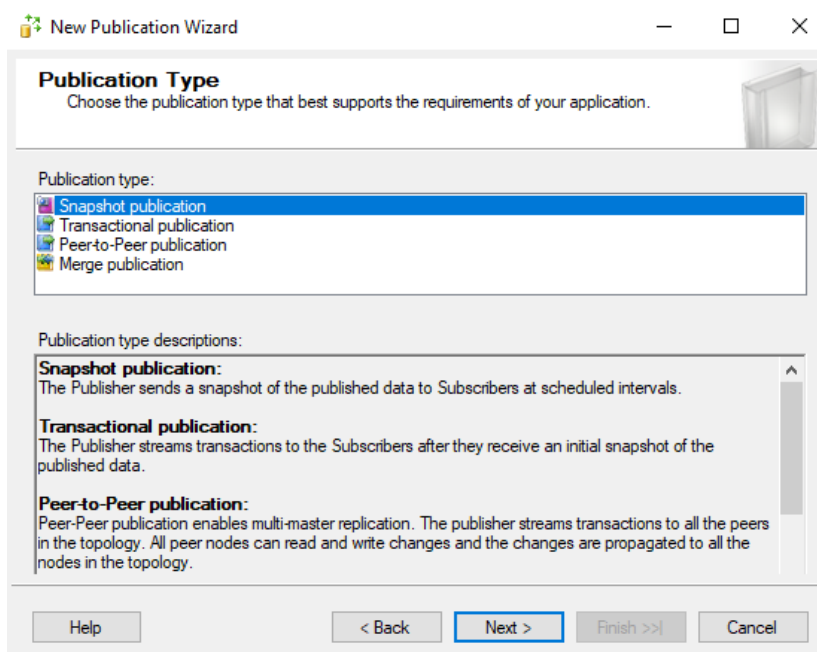


2ª Fase Relatório Técnico – Complementos de Bases de Dados

Pressionando “Next >” somos encaminhados para a janela de seleção de base de dados para publicação. Sendo que a base de dados *AdventureNewData* consiste na nossa base de dados principal (sede/North America), escolhemos esta base de dados como a que “publica os dados” para as outras regiões (Europa e Pacífico).

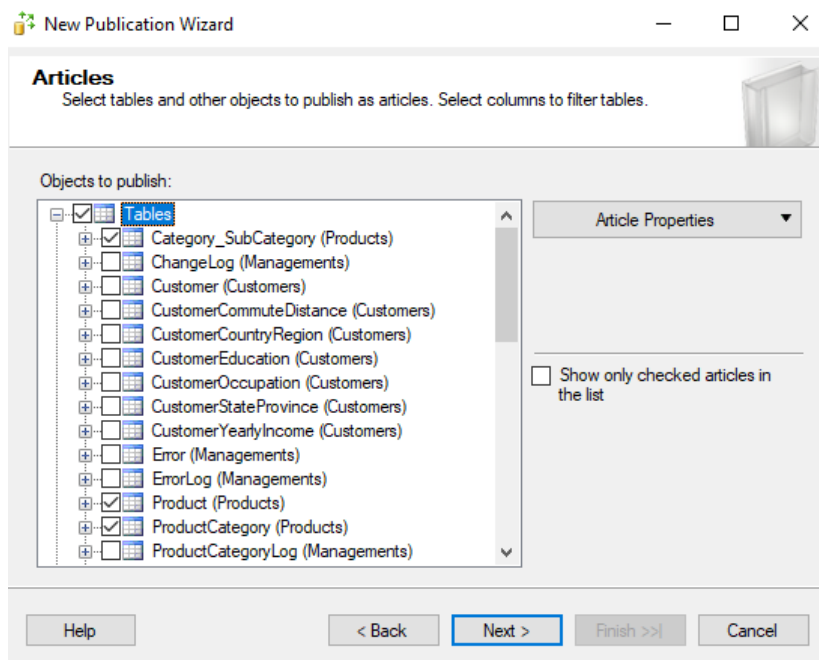


Posteriormente, tivemos que selecionar o tipo de publicação que desejaríamos nesta criação de publicação. Optamos pela publicação snapshot, que iremos explicar o porquê no fim da explicação do procedimento.

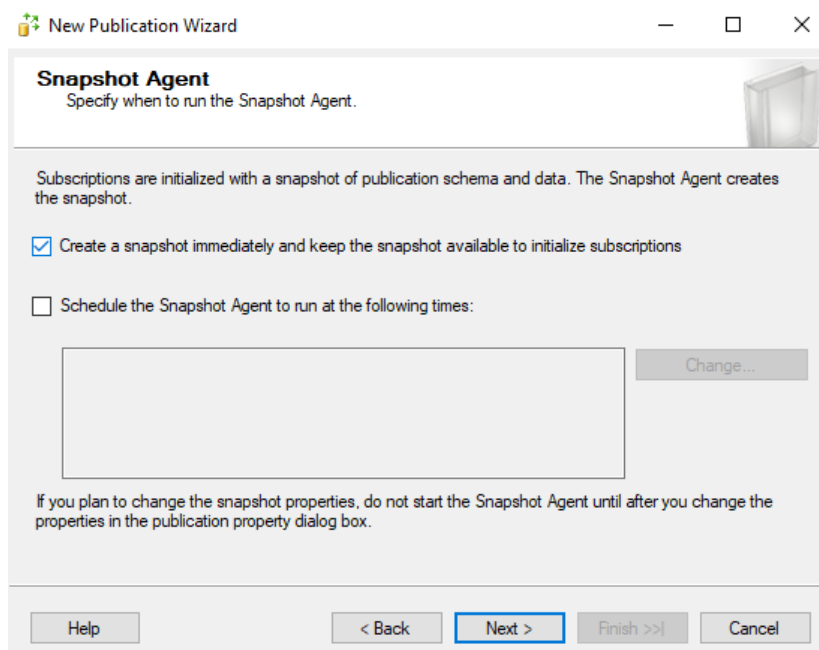


O próximo passo, consistia na seleção das tabelas que seriam replicadas da sede para as outras regiões existentes. Como era pedido no enunciado do projeto, era apenas necessária informação sobre o catálogo de produtos e promoções, pelo que selecionamos todos os componentes pertencentes ao Schema “Products”.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

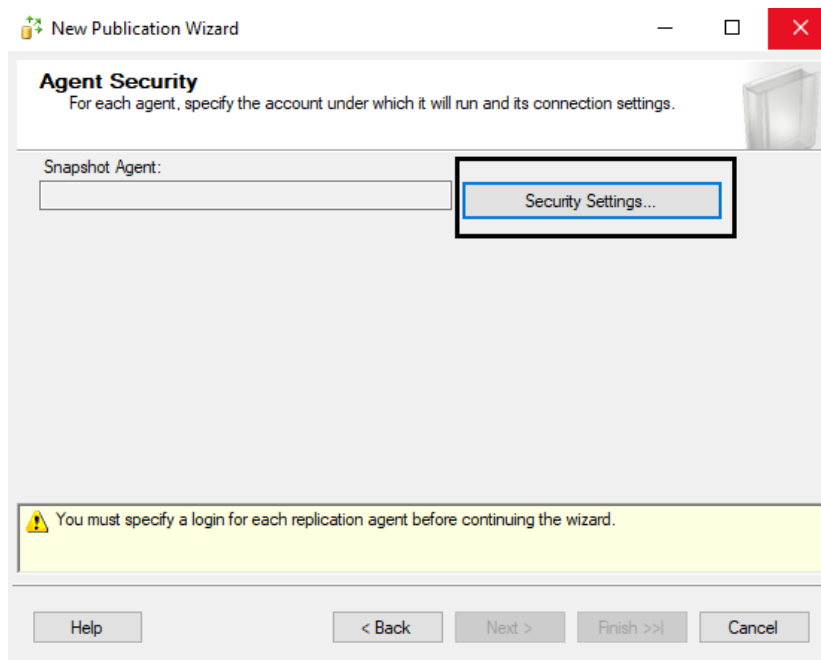


O passo seguinte, consistia apenas na criação imediata do snapshot no Snapshot Agent, seguida de um clique em “Next >”.

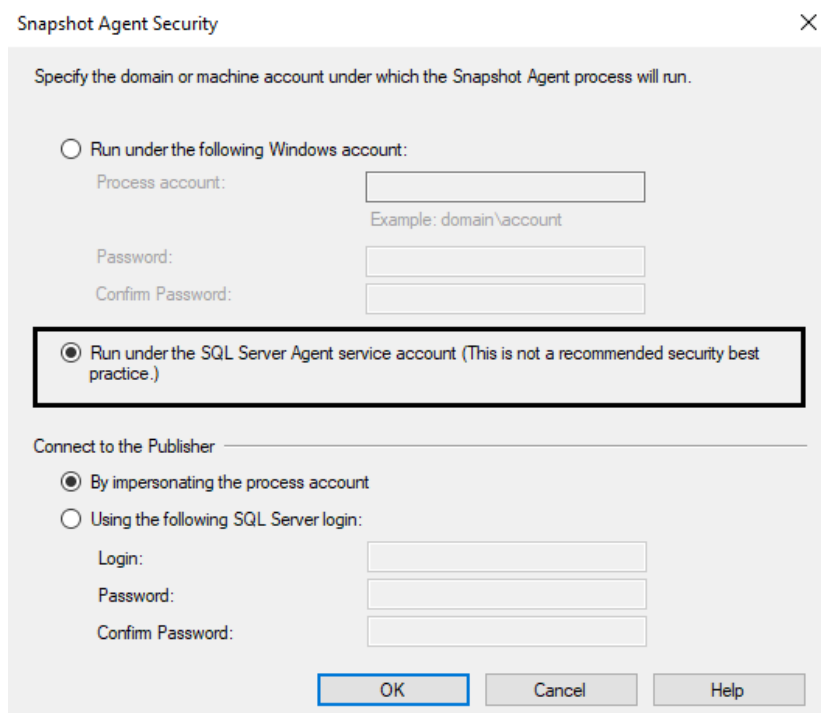


Após o passo anterior fomos às “Security Settings”.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

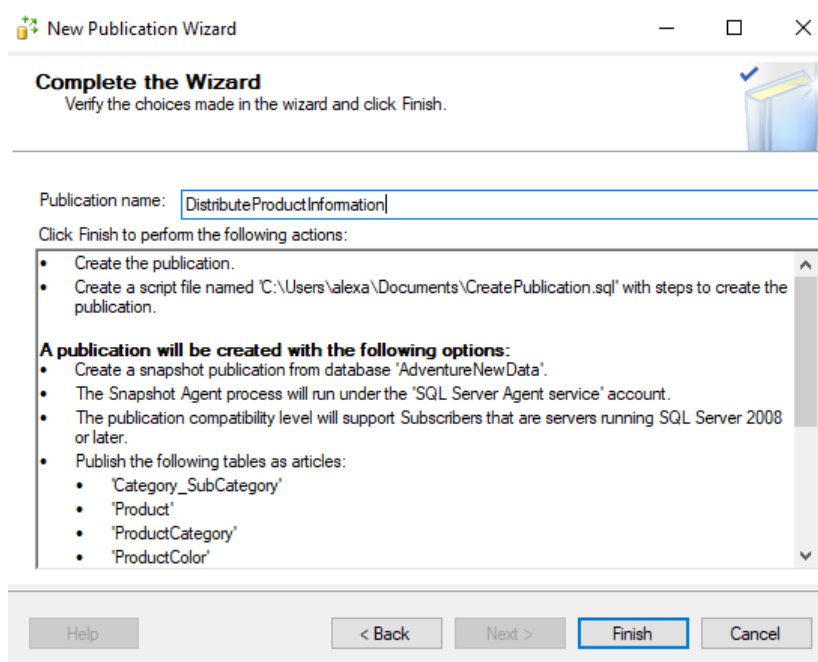


E aqui seleccionamos a execução com o SQL Server Agent, do qual ocorreram alguns problemas devido à configuração de contas do Windows, mas foi posteriormente resolvido com recurso ao Job Activity Monitor

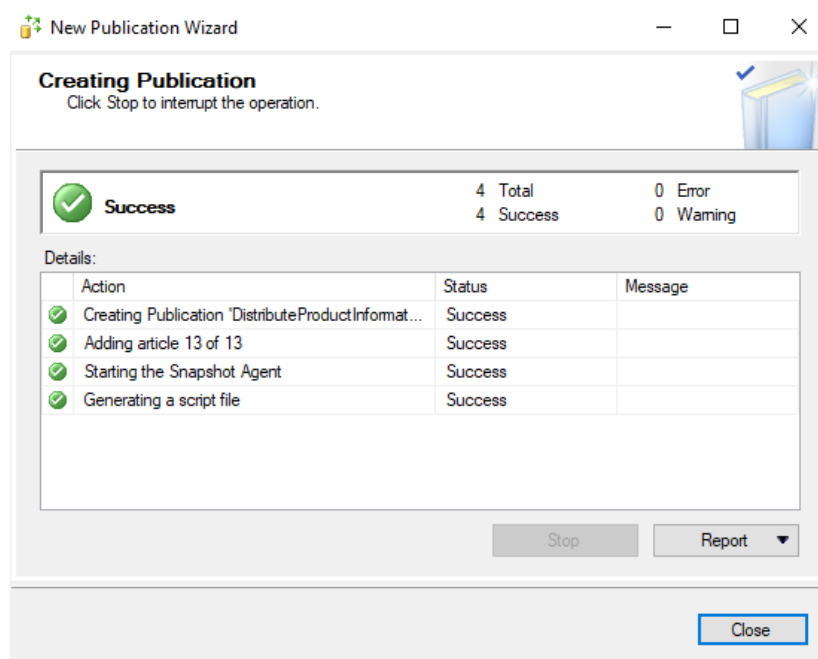


Por fim, apenas era necessário escolher o nome da nossa nova publicação.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

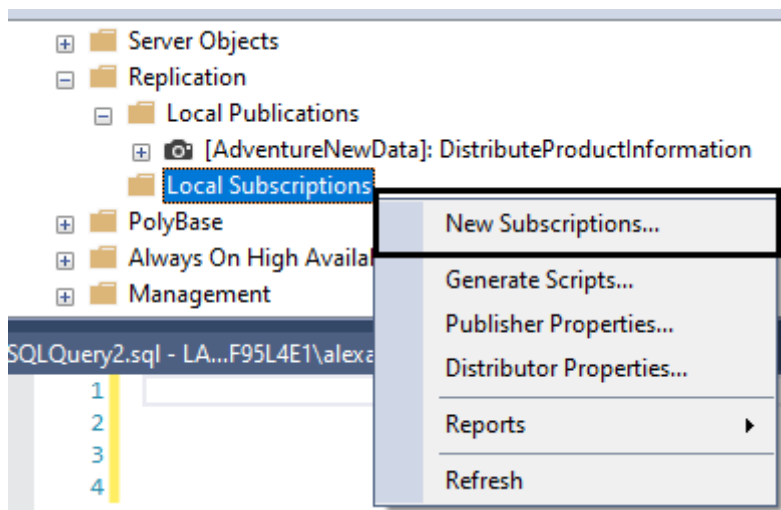


E concluímos a configuração do publisher.

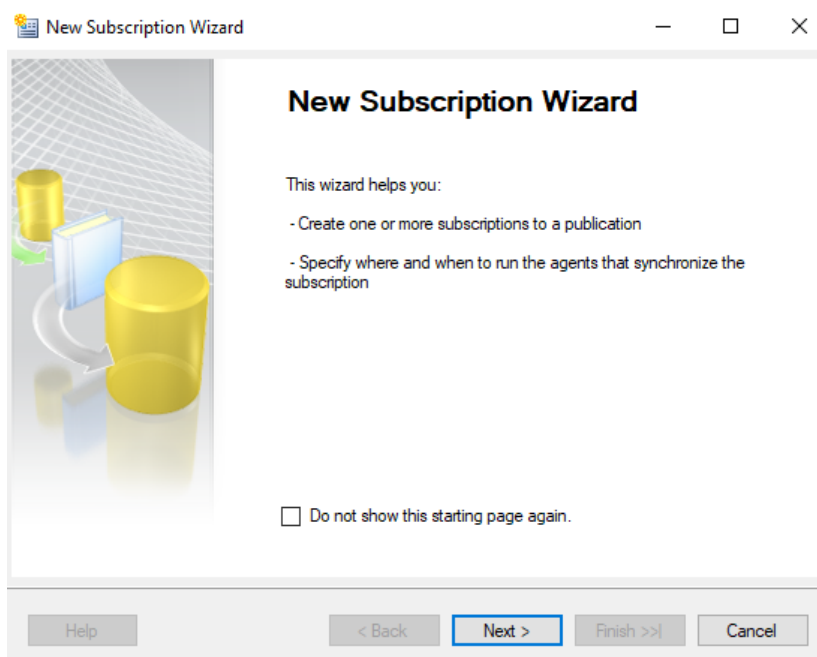


Criação de um subscriber

Depois da criação de um publisher, , processo este que se inicia com um clique com o botão direito do rato em “Local Subscription” (que se encontra dentro da pasta “Replication”). Posteriormente, seleccionamos a opção “New Subscriptions...”.

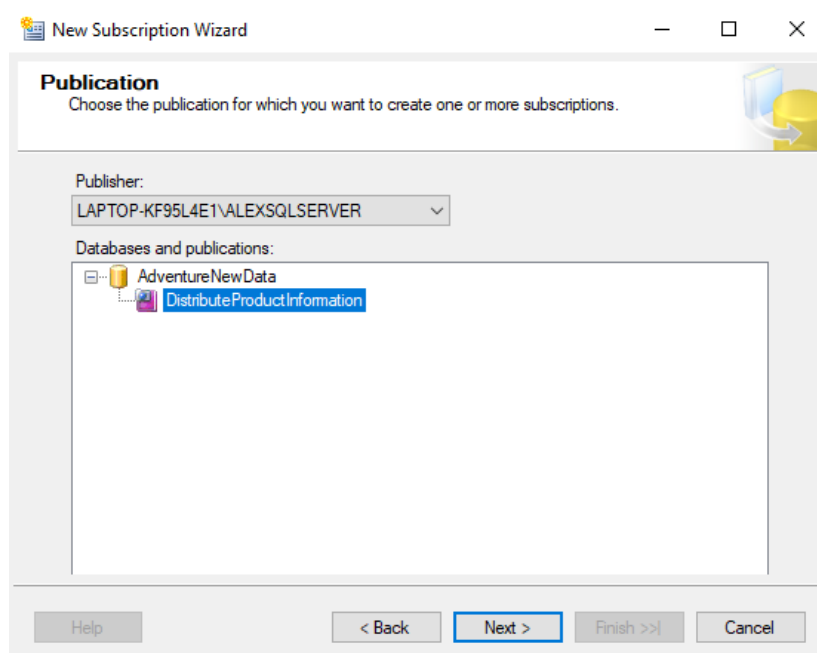


Após o último passo somos encaminhados para o “New Subscription Wizard”, onde se inicia a configuração do publisher.

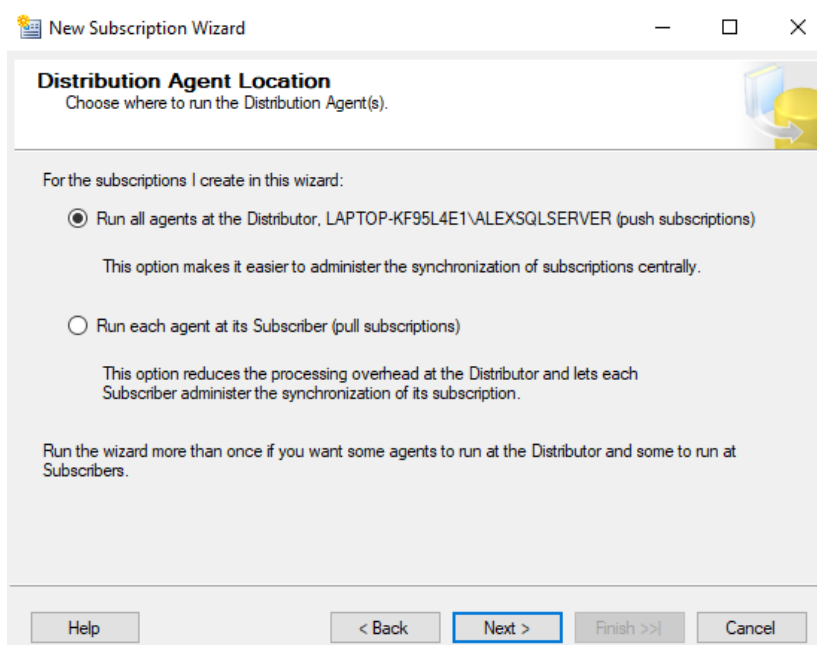


Posteriormente, tivemos que seleccionar a publicação, de onde desejávamos replicar os dados (cuja criação se encontra referida anteriormente).

2ª Fase Relatório Técnico – Complementos de Bases de Dados

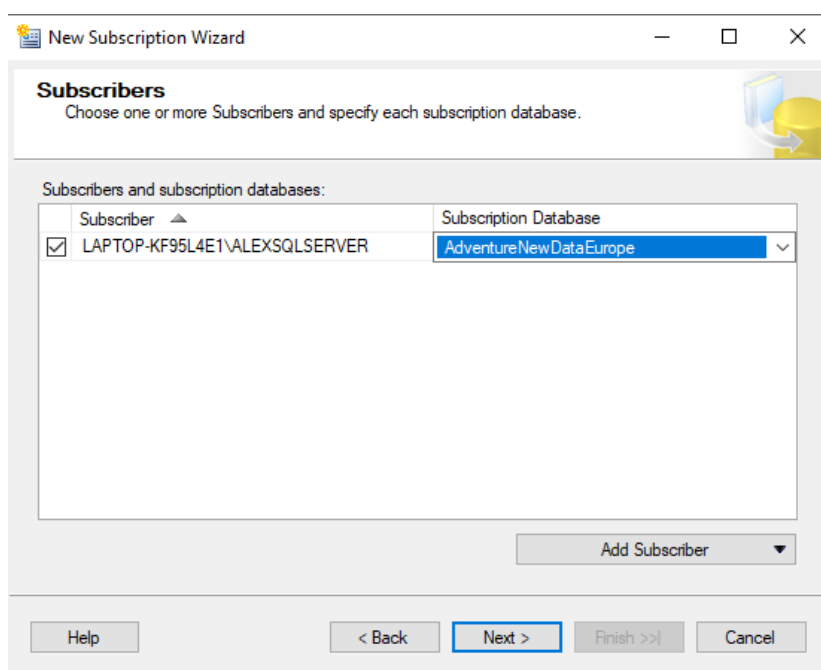


Depois, seleccionámos push subscriptions e pressionamos “Next >”.

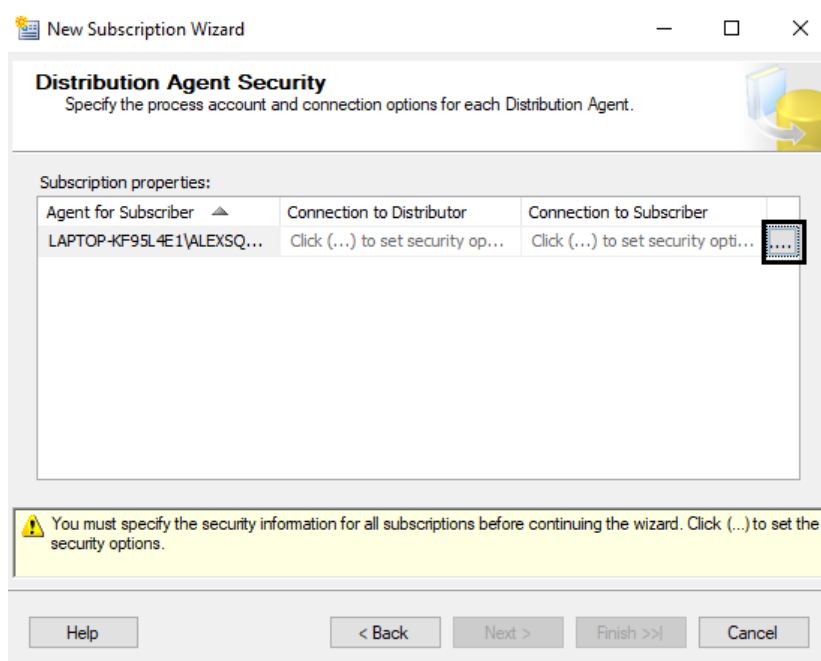


Após termos pressionado “Next >”, seguimos para a seleção da base de dados de subscrição, onde seleccionamos as bases de dados criadas para as regiões existentes, Europa e Pacífico.

2ª Fase Relatório Técnico – Complementos de Bases de Dados



De seguida, após a janela anterior, seguimos para os Security settings, através de um clique no botão com "...".



E aqui seleccionamos a execução com o SQL Server Agent, do qual ocorreram alguns problemas devido à configuração de contas do Windows, mas foi posteriormente resolvido com recurso ao Job Activity Monitor.

Distribution Agent Security ✕

Specify the domain or machine account under which the Distribution Agent process will run when synchronizing this subscription.

☐ Run under the following Windows account:

Process account:
Example: domain\account

Password:
Confirm Password:

☒ Run under the SQL Server Agent service account (This is not a recommended security best practice.)

Connect to the Distributor

☒ By impersonating the process account

☐ Using a SQL Server login

The connection to the server on which the agent runs must impersonate the process account. The process account must be a member of the Publication Access List.

Connect to the Subscriber

☒ By impersonating the process account

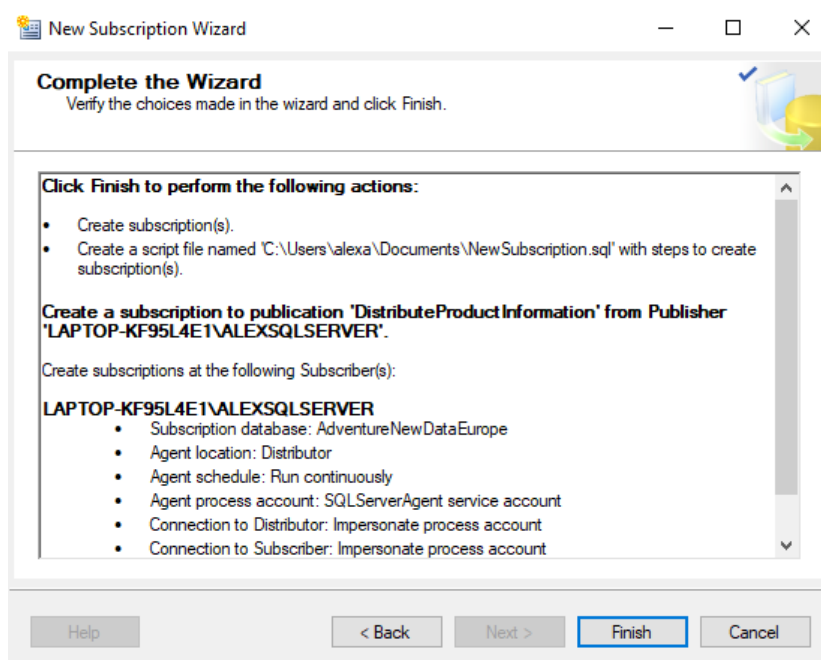
☐ Using the following SQL Server login:

Login:
Password:
Confirm password:

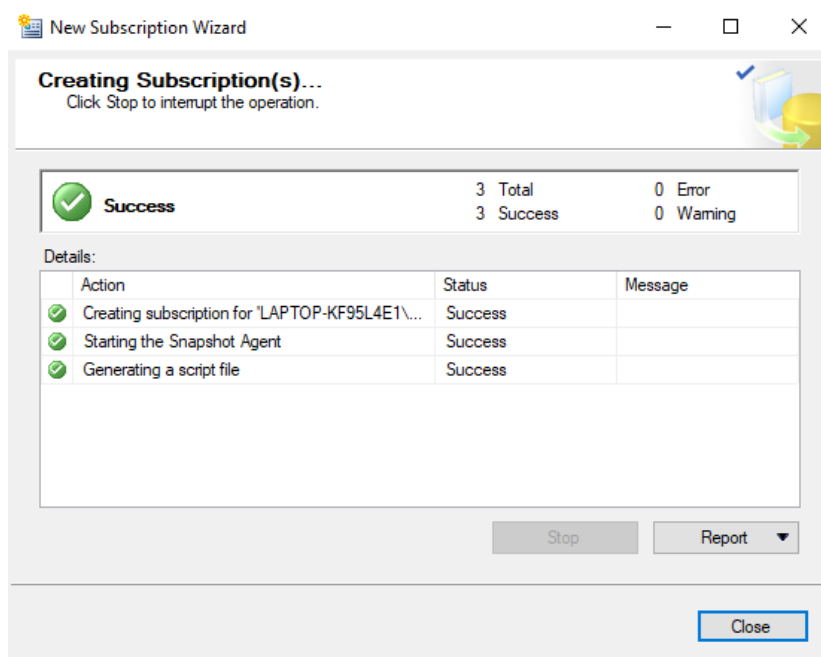
The login used to connect to the Subscriber must be a database owner of the subscription database.

Neste penúltimo ecrã, apenas necessitamos de pressionar “finish”, depois de devida confirmação de conformidade na criação.

2ª Fase Relatório Técnico – Complementos de Bases de Dados

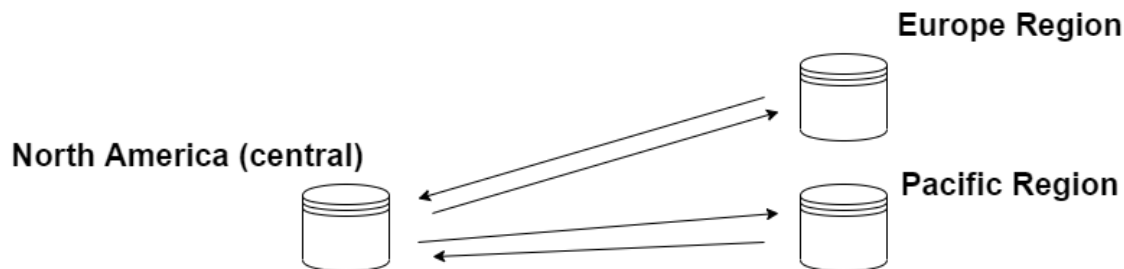


E finalizamos a criação do subscriber.



Métodos de replicação abordados

Partitioning by region // Bidiretional

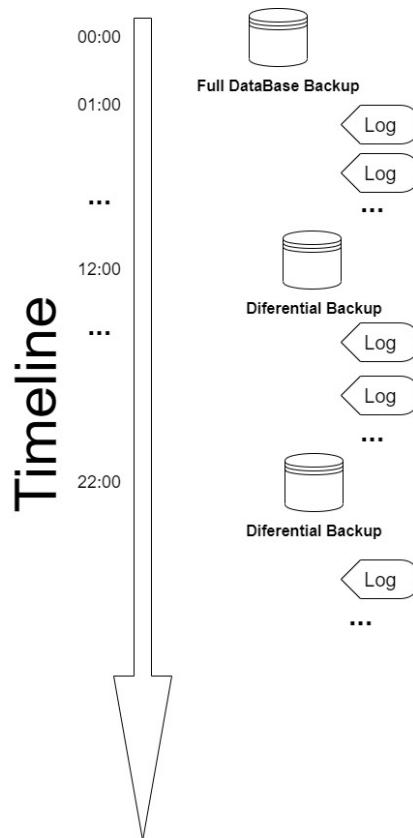


O método abordado na replicação que usamos como exemplo, retrata a ligação North America para as outras regiões (North America -> Europe Region, Pacific Region). Não contemos um nível aplicacional onde podemos retratar uma simulação de servidores, mas o objectivo era de partição por região, sendo que a região da Europa e do Pacífico, têm acesso aos dados da central. Foi escolhida uma replicação “snapshot”, devido a ser uma replicação de dados de leitura, com poucas transações, devido a ser dados relativos a produtos, e não existe uma grande adição ou remoção de produtos.

Quanto à direção das regiões á central, optamos por uma replicação transaccional, pois os dados a replicar, constituem-se de vendas com promoções, onde ocorre um nível substancial de transações. As promoções limitam um pouco a quantidade de tráfego que a base de dados têm, mas não delimita o trafego geral, pois não deixam de ser dados relativos a vendas.

16. Política de Backups

Quanto ao nosso plano de Backups, definimos a seguinte estrutura:



Sendo que os nossos Full Backups estão planeados para a 00:00(UTC), os Full com Transactional Log são feitos hora a hora e os diferenciais são executados duas vezes ao dia, uma ao 12:00, outra as 22:00, previamente ao full que é efetuado a 00:00 diariamente (como referido anteriormente).

No caso de ocorrer um erro, deverá ser executado o backup de tail.

Temos também backups parciais que guardam todos os files e filegroups destinados a cada schema. Deve ser feito um backup destes quando é efetuada um grande tráfico de operações a um determinado schema.

Os backups regulares serão armazenados em NAS (Network Attached Storage), sendo que mensalmente os backups integrais serão movidos para um armazenamento terciário, tal como um DVD, e depois todos esses backups serão removidos da NAS de forma a otimizar os custos do armazenamento.

17. Mongo DB

Nesta parte foi criada uma tabela ProductComment para armazenar os comentários dos produtos, que foram gerados através de um procedure ([Products].populateComments). Novos comentários podem ser criados através do procedure ([Products].newComment). Os queries das etapas do enunciado de MongoDB encontram-se no ficheiro **Querys MongoDB.js**.

18. Modificações da 1ª Fase

Foram adicionados alguns triggers que eram pedidos na 1ª fase do projeto, foram alterados alguns procedures da programability das sales e foram corrigidos alguns problemas da criação da base de dados AdventureNewData.

19. Descrição da Demonstração

A pasta (ProjectoGrupo6) contém todos os backups efetuados e os filegroups da criação das bases de dados e respetiva tabelas. Para uma correta execução dos scripts relativos a filegroups, criação de bases de dados e tabelas, ou criação de backups, esta pasta deverá ser posicionada no disco C:\. De momento não é necessário uma execução, pois já se encontram dentro da pasta os ficheiros necessários, gerados anteriormente. Sendo que já existiu uma avaliação dos filegroups da sede (AdventureNewData), estes não tem qualquer alteração e não estão na pasta anexada (ProjectoGrupo6) devido às complicações das permissões de administrador. Caso o professor queira verificar se a criação dos files e filegroups da base de dados sede estão em conformidade, deverá proceder à colocação da pasta no disco C:\ e executar o ficheiro 1-Files & Filegroups.

1. Fazer import do ficheiro AdventureOldData.bak
2. Correr o ficheiro 1 - Files & Filegroups.sql;
3. Correr o ficheiro 2 - Create Tables.sql;
4. Correr o ficheiro 3 - Data Migration.sql;
5. Correr o ficheiro 3.1 - Populate Comment Table.sql;
6. Correr o ficheiro 3.2 - Migration MongoDB.sql;
7. Correr o ficheiro 4 - Data Migration (Control Views).sql;
8. Correr o ficheiro 5 - Admin Account Procedures;
9. Correr o ficheiro 6 - Management Procedures.sql;
10. Correr o ficheiro 7 - Generator Procedure.sql;
11. Correr o ficheiro 8 - Programability - Sales;
12. Correr o ficheiro 9 - Triggers;
13. Correr o ficheiro 10 – Indexes;
14. Correr o ficheiro 11 - Distribution and Replication - Files & Filegroups;
15. Correr o ficheiro 12 - Distribution and Replication - Table Creation;
16. Correr o ficheiro 13 - Access & Privileges;
17. Correr o ficheiro 14 - Access & Privileges – Procedures;
18. Correr o ficheiro 15 - Backups Configuration;