

NEO Monitoring

Modelação e Implementação de Base de Dados

Nome do Estudante

Licenciatura em Engenharia Informática

Unidade Curricular: Base de Dados

Ano Letivo 2025/2026

Conteúdo

1	Modelo Entidade–Relacionamento do Sistema <i>NEO Monitoring</i>	2
1.1	Introdução	2
1.2	Visão geral do modelo	2
1.3	Diagramas ER	3
1.4	Entidades	3
1.4.1	Asteroide	3
1.4.2	Classe Orbital	3
1.4.3	Solução Orbital	3
1.4.4	Aproximação Próxima	4
1.4.5	Prioridade Alerta e Nível Alerta	4
1.4.6	Alerta	4
1.4.7	Entidades do contexto observacional	4
1.5	Relacionamentos e cardinalidades	4
1.6	Conclusão	5
2	Esquema Relacional	6
2.1	Resumo do esquema	6
3	Implementação SQL e <i>Views</i>	7
3.1	Introdução	7
3.2	Estrutura geral do código SQL	7
3.3	<i>Views</i> definidas	8
3.4	Exemplos de pesquisas suportadas	8
3.4.1	Últimos 5 asteroides detectados	8
3.4.2	Top 10 PHAs de maior dimensão	9
3.4.3	Centros com mais observações	9
3.4.4	Observações com contexto completo	9
3.5	Conclusão	9

Capítulo 1

Modelo Entidade–Relacionamento do Sistema *NEO Monitoring*

1.1 Introdução

O presente capítulo descreve, de forma estruturada e em linguagem académica, o modelo de dados conceptual desenvolvido para o sistema *NEO Monitoring*, responsável pela monitorização de objectos próximos da Terra (*Near Earth Objects* – NEOs) e pela gestão de observações e alertas associados.

O modelo é especificado através de um diagrama Entidade–Relacionamento (ER), construído na ferramenta ERDPlus-old. Neste diagrama são representadas as entidades relevantes, os respectivos atributos e os relacionamentos entre elas, incluindo as cardinalidades mínimas e máximas. A partir deste modelo conceptual será posteriormente obtido o esquema relacional implementado em *SQL Server*.

1.2 Visão geral do modelo

O modelo ER pode ser organizado em três áreas principais:

- **Núcleo orbital:** entidades relacionadas com a caracterização física e orbital de cada asteroide, como **Asteroide**, **Classe Orbital**, **Solução Orbital** e **Aproximação Próxima**.
- **Gestão de alertas:** entidades responsáveis pela classificação do risco associado aos asteroides, nomeadamente **Alerta**, **Prioridade Alerta** e **Nível Alerta**.
- **Contexto observacional:** entidades que representam o processo de observação astronómica, como **Observação**, **Imagem**, **Astronomo**, **Centro Observação**, **Equipamento** e **Software**.

1.3 Diagramas ER

1.4 Entidades

Nesta secção descrevem-se as principais entidades do modelo, com os seus atributos mais relevantes. A lista completa de atributos pode ser consultada no capítulo do esquema relacional.

1.4.1 Asteroide

A entidade **Asteroide** representa cada objecto celeste monitorizado pelo sistema.

Os atributos incluem identificadores externos (por exemplo `id_csv`, `id_spk`), nomes (`nome_completo`, `nome_asteroide`, `pdes`) e propriedades físicas como magnitude absoluta, diâmetro, incerteza do diâmetro, albedo e data de descoberta.

1.4.2 Classe Orbital

A entidade **Classe Orbital** define as classes em que uma órbita pode ser enquadrada (por exemplo, Aten, Apollo, Amor), através de um código de classe e de uma descrição.

1.4.3 Solução Orbital

A entidade **Solução Orbital** representa uma solução específica para a órbita de um asteroide, válida para um determinado intervalo temporal. Para um mesmo asteroide podem existir várias soluções, correspondentes a refinamentos sucessivos dos elementos orbitais.

Os seus atributos incluem:

- elementos de época (por exemplo `epoca_jd`, `epoca_mjd`, `epoca_cal`, `equinocio`);
- elementos orbitais (excentricidade, semi-eixo maior, perielio, inclinação, argumentos e longitudes angulares, período orbital, distâncias MOID);
- incertezas associadas a cada parâmetro;
- indicadores de qualidade e proveniência (`rms`, `codigo_condicao`, `referencia_mpc`, etc.);
- período de validade e indicação se se trata da solução actual.

1.4.4 Aproximação Próxima

A entidade **Aproximação Próxima** regista aproximações significativas de um asteroide à Terra, com data/hora, distâncias (em unidades astronómicas e distâncias lunares), velocidade relativa e indicação se a aproximação é considerada crítica.

1.4.5 Prioridade Alerta e Nível Alerta

Prioridade Alerta define níveis qualitativos (Alta, Média, Baixa) e **Nível Alerta** define níveis quantitativos (por exemplo 1 a 4), associados a intervalos de diâmetro, valores de MOID e erros máximos permitidos.

1.4.6 Alerta

A entidade **Alerta** representa um evento de alerta gerado pelo sistema relativamente a um determinado asteroide, incluindo o código da regra que o originou, datas de geração, reconhecimento e encerramento, bem como um título e uma descrição.

1.4.7 Entidades do contexto observacional

As entidades **Centro Observação**, **Equipamento**, **Astronomo**, **Software**, **Observação** e **Imagem** modelam o contexto operacional: onde e como os dados são adquiridos, quem os obtém, com que instrumentos e que ficheiros de imagem resultam dessas observações.

1.5 Relacionamentos e cardinalidades

Nesta secção são descritos, em termos qualitativos, os relacionamentos principais do modelo.

Entre o núcleo orbital e a gestão de alertas salientam-se, por exemplo:

- **Tem soluções (Asteroide–Solução Orbital)**: um asteroide pode ter várias soluções orbitais, e cada solução orbital pertence a um único asteroide.
- **Tem aprox. (Asteroide–Aproximação Próxima)**: um asteroide pode ter várias aproximações registadas, cada aproximação é de um único asteroide.
- **Alvo de (Asteroide–Alerta)**: cada alerta é sempre relativo a um asteroide; um asteroide pode ser alvo de vários alertas.
- **Classifica Prioridade / Classifica nível (Prioridade Alerta, Nível Alerta–Alerta)**: um alerta tem obrigatoriamente uma prioridade e, opcionalmente, um nível quantitativo.

No contexto observacional destacam-se:

- **É Observado (Asteroide–Observacao):** cada observação refere-se a um único asteroide, que pode ter nenhuma ou várias observações.
- **Realiza (Astronomo–Observacao):** cada observação é realizada por um astrônomo.
- **Processa (Software–Observacao) e Utilizado em (Equipamento–Observacao):** cada observação utiliza exactamente um equipamento e um software registado.
- **Produz imagens (Observacao–Imagem) e Tem imagens (Asteroide–Imagem):** uma observação pode produzir várias imagens, cada imagem pertence a um asteroide.

1.6 Conclusão

O modelo Entidade–Relacionamento apresentado garante uma representação coerente e detalhada do domínio de aplicação do sistema *NEO Monitoring*, preparando o terreno para a definição do esquema relacional e para a implementação física da base de dados.

Capítulo 2

Esquema Relacional

Neste capítulo é apresentada a tradução do modelo ER para o modelo relacional, utilizando a notação clássica de esquemas de tabelas do tipo:

TABELA(chave_primária, *chaves_estrangeiras*, outros_atributos)

2.1 Resumo do esquema

A título de exemplo, alguns dos esquemas mais importantes são:

- **ASTEROIDE**(id_asteroide, id_csv, id_spk, nome_completo, pdes, nome_asteroide, prefixo, flag_neo, flag_pha, H_mag, diametro_km, diametro_sigma_km, albedo, data_descoberta)
- **CLASSE_ORBITAL**(codigo_classe, descricao)
- **SOLUCAO_ORBITAL**(id_solucao_orbital, *id_asteroide*, *codigo_classe*, ...)
- **APROXIMACAO_PROXIMA**(id_aproximacao_proxima, *id_asteroide*, *id_solucao_orbital*, ...)
- **ALERTA**(id_alerta, *id_asteroide*, *id_solucao_orbital*, *id_aproximacao_proxima*, *id_prioridade_alerta*, *id_nivel_alerta*, ...)
- **OBSERVACAO**(id_observacao, *id_asteroide*, *id_astronomo*, *id_equipamento*, *id_software*, ...)
- **IMAGEM**(id_imagem, *id_asteroide*, *id_observacao*, ...)

Nesta notação:

- atributos sublinhados representam chaves primárias;
- atributos em itálico representam chaves estrangeiras;
- os restantes são atributos não chave.

Capítulo 3

Implementação SQL e *Views*

3.1 Introdução

Este capítulo descreve a implementação da base de dados do sistema *NEO Monitoring* em *SQL Server*, bem como o conjunto de *views* criadas para apoiar as pesquisas mais comuns sobre asteroides, aproximações próximas, alertas e observações.

Partindo do esquema relacional definido no capítulo anterior, foi desenvolvido um conjunto de instruções SQL para criação das tabelas e das respectivas restrições de integridade (chaves primárias e estrangeiras). Sobre esta estrutura física foram ainda definidas diversas *views*, que encapsulam consultas típicas e simplificam o acesso à informação por parte da aplicação.

3.2 Estrutura geral do código SQL

O código SQL de criação da base de dados encontra-se organizado em blocos `CREATE TABLE` com as respectivas definições de `PRIMARY KEY` e `FOREIGN KEY`, respeitando as dependências entre tabelas.

De forma resumida:

- As entidades principais do modelo são implementadas como tabelas em *SQL Server* (**Asteroide**, **Solucao_Orbital**, **Aproximacao_Proxima**, **Alerta**, **Observacao**, **Imagem**, **Centro_Observacao**, **Equipamento**, **Astronomo**, **Software**, **Prioridade_Alerta**, **Nivel_Alerta**).
- As chaves primárias são do tipo `INT IDENTITY(1,1)` ou `VARCHAR` (no caso de códigos externos), e as chaves estrangeiras são declaradas explicitamente com `FOREIGN KEY`.
- A nulidade dos atributos reflecte a obrigatoriedade ou opcionalidade das participações nos relacionamentos: por exemplo, `Alerta.id_asteroide` é `NOT NULL`, enquanto `Alerta.id_solucao_orbital` é opcional.

O código completo de criação das tabelas pode ser consultado em anexo.

3.3 *Views* definidas

Para simplificar as consultas mais frequentes, foram criadas várias *views* em *SQL Server*. Estas *views* funcionam como “tabelas virtuais” e encapsulam **SELECTs** com **JOINS** e filtros, permitindo que a aplicação acesse a informação complexa através de instruções simples.

As principais *views* são:

- **vw_Ultimos5AsteroidesDetetados**: devolve os cinco asteroides mais recentemente descobertos;
- **vw_AsteroidesNEO**, **vw_AsteroidesPHA**, **vw_AsteroidesNEOePHA**: filtram asteroides por tipo (NEO, PHA ou ambos);
- **vw_CentrosComMaisObservacoes**: calcula o número de observações por centro;
- **vw_RankingAsteroidesPHA_MaiorDiametro**: fornece um ranking dos PHAs ordenados por diâmetro;
- **vw_Observacoes_Completo**: junta observações com asteroide, astrónomo, centro, equipamento e software;
- **vw_Alertas_Ativos_Detalhe**: lista alertas activos com informação de prioridade, nível e asteroide;
- **vw_Asteroide_OrbitalAtual**: associa cada asteroide à sua solução orbital marcada como actual.

3.4 Exemplos de pesquisas suportadas

Alguns exemplos de consultas que a aplicação poderá executar, recorrendo a estas *views*, são:

3.4.1 Últimos 5 asteroides detectados

```
SELECT *  
FROM    vw_Ultimos5AsteroidesDetetados;
```

3.4.2 Top 10 PHAs de maior dimensão

```
SELECT *  
FROM    vw_RankingAsteroidesPHA_MaiorDiametro  
WHERE   posicao_ranking <= 10  
ORDER BY posicao_ranking;
```

3.4.3 Centros com mais observações

```
SELECT TOP (10)  
    id_centro,  
    nome,  
    pais,  
    cidade,  
    total_observacoes  
FROM    vw_CentrosComMaisObservacoes  
ORDER BY total_observacoes DESC;
```

3.4.4 Observações com contexto completo

```
SELECT TOP (50)  
    datahora_observacao,  
    nome_asteroide,  
    nome_astronomo,  
    nome_centro,  
    nome Equipamento,  
    nome Software,  
    modo,  
    magnitude  
FROM    vw_Observacoes_Completo  
ORDER BY datahora_observacao DESC;
```

3.5 Conclusão

As *views* desenvolvidas constituem uma camada de abstracção importante entre a base de dados e a aplicação, simplificando as consultas mais frequentes e facilitando a manutenção do código SQL. Em conjunto com o esquema relacional, formam a base sólida sobre a qual será construída a aplicação de monitorização e análise de NEOs.