



**PREFEITURA  
BELO HORIZONTE**

**Empresa de Informática e Informação de Belo Horizonte S/A**

**PRODABEL**

**PROJETO GEOPROCESSAMENTO CORPORATIVO**

# **Guia de boas práticas para o gerenciamento de dados vetoriais no Oracle Spatial**

**Belo Horizonte, Julho de 2013**

## Conteúdo

Sobre este documento .....	4
Público-alvo .....	4
1. Introdução .....	5
2. Modelagem de dados espaciais.....	6
3. Dados vetoriais no SDO_GEOMETRY.....	6
4. Metadados espaciais .....	7
Número de dimensões .....	7
Limites de cada dimensão .....	7
Tolerância .....	7
Sistema de referência espacial .....	7
Views e tabela de metadados espaciais .....	7
5. Validação do formato do dado espacial.....	8
Funções de validação.....	8
SDO_GEOM.VALIDATE_GEOMETRY_WITH_CONTEXT .....	8
SDO_GEOM.VALIDATE_LAYER_WITH_CONTEXT.....	8
SDO_GEOMETRY.ST_ISVALID .....	8
Critérios de validação .....	9
Erros mais comuns em geometrias inválidas .....	9
Funções para correção de geometrias inválidas .....	10
SDO_UTIL.REMOVE_DUPLICATE_VERTICES .....	10
SDO_UTIL.RECTIFY_GEOMETRY.....	10
SDO_UTIL.EXTRACT .....	10
SDO_UTIL.GETNUMELEM .....	10
SDO_UTIL.GETNUMVERTICES e SDO_UTIL.GETVERTICES .....	10
6. Indexação espacial.....	11
Parâmetros .....	11
tablespace.....	11
work_tablespace .....	11
layer_gtype .....	11
sdo_indx_dims.....	12
7. Análise espacial .....	12
Operadores espaciais .....	12
ORDERED hint.....	13

Funções espaciais .....	13
8. Recomendações finais .....	13

## **Sobre este documento**

Este guia de boas práticas é parte do esforço em auxiliar e aprimorar o desenvolvimento e suporte a sistemas e aplicações que fazem uso de dados espaciais armazenados no Sistema Gerenciador de banco de Dados – SGBD – Oracle através dos recursos espaciais Locator e Spatial.

Este documento não é um manual de Locator ou de Spatial e não pretende ensinar como licenciar, instalar, e utilizar seus recursos.

Por ser um guia de boas práticas, este documento provê orientações para a boa utilização de alguns recursos do Locator e do Spatial definidos como essenciais ou importantes para a melhor utilização e gerenciamento de dados espaciais no Oracle.

## **Público-alvo**

As recomendações contidas neste documento destinam-se a todos profissionais de tecnologia da informação – TI – da PBH, mas especialmente àqueles que sejam familiarizados com os SGBD Oracle, com os recursos espaciais Locator e Spatial, e que já utilizam ou pretendem utilizar informações geográficas em suas aplicações. Para melhor aproveitamento das recomendações, espera-se do leitor conhecimentos prévios em projetos de Banco de Dados, PL/SQL e noções básicas de Geoprocessamento.

## 1. Introdução

O Oracle agrega características espaciais ao SGBD Oracle através dos recursos Locator e Spatial. O Locator provê recursos básicos para gerenciamento de dados espaciais no banco de dados, já o Spatial provê todos os recursos do Locator adicionado de recursos avançados para soluções mais sofisticadas e complexas de Sistemas de Informações Geográficas.

De todos os recursos que o Locator e Spatial dispõem, existem aqueles mais comuns que consideramos essenciais para o gerenciamento de dados vetoriais no SGBD Oracle. São eles:

### **O tipo de dado SDO\_GEOMETRY**

O SDO\_GEOMETRY é tipo espacial nativo do Oracle para armazenar dados espaciais. É um objeto com vários atributos e métodos para armazenar e recuperar as informações do dado espacial tais como o tipo de geometria, o sistema de referência espacial do dado e as coordenadas que compõem os vértices do desenho da geometria. O bom entendimento das suas características é essencial para a boa utilização dos seus recursos.

### **Os metadados espaciais**

Os metadados espaciais são tabelas e views de metadados que registram as informações que definem uma camada espacial no banco de dados. Seu preenchimento deve ser obrigatório e as informações devem ser escolhidas com o conhecimento adequado sobre o dado espacial.

### **A validação do dado espacial**

A validação é uma funcionalidade do Oracle que permite verificar se a geometria de um dado espacial foi desenhada corretamente ou não. É muito importante para garantir a qualidade do dado e o bom funcionamento das consultas espaciais.

### **A indexação espacial**

A indexação impacta diretamente no desempenho das consultas sendo obrigatórias para a utilização de alguns operadores espaciais. A atribuição de alguns parâmetros na criação dos índices pode melhorar seu desempenho.

### **Os operadores para análise espacial**

Os operadores espaciais são utilizados nas consultas para verificar o relacionamento espacial entre dados.

Estes recursos devem ser explorados em sua plenitude de maneira a proporcionar ao usuário final a melhor experiência na qualidade das informações e no desempenho das consultas.

Para melhor aproveitamento destes recursos, também se deve adotar algumas recomendações para modelagem de dados espaciais.

Todos esses assuntos são abordados com maior profundidade nas próximas seções.

## 2. Modelagem de dados espaciais

A modelagem influi diretamente nas tabelas a serem criadas e na eficiência do banco de dados, pois é ela quem determina as possibilidades de cruzamento de dados a serem exploradas para a recuperação das informações.

A criação do modelo de dados espaciais deve receber a atenção e dedicação proporcionais à sua importância. É na modelagem que são definidos os atributos e relacionamentos entre as tabelas. Mesmo sendo independente da solução de banco de dados espacial adotada, é importante ressaltar algumas recomendações para modelagem de dados espaciais para a criação de um banco de dados eficiente:

**Ao se modelar uma tabela espacial normalizada, a relação entre o atributo espacial e os outros atributos da tabela deve sempre ser 1:1.**

Não deve haver réplicas de dados espaciais em uma mesma tabela. Isso dificulta a manutenção do dado e onera consultas espaciais, que já são custosas, gerando retrabalho na verificação de dados iguais.

**Um atributo espacial deve ter um único formato de dado espacial (ponto, linha, polígono,...) definido para ele.**

Um atributo deve ter apenas uma representação. Para isso, o tipo coleção, adotado em alguns bancos de dados e que permite que múltiplos formatos espaciais sejam armazenados na tabela, deve ser evitado sempre que possível. Caso haja a possibilidade de ter mais de uma representação para o dado cada formato deve ter um atributo próprio.

**Dados espaciais que não compartilham os mesmos atributos devem ser separados em tabelas diferentes.**

Dados que não compartilham os mesmo atributos devem estar em tabelas separadas, pois não representam a mesma entidade/classe, ainda que tenham a mesma representação e formato espacial. Uma análise mais minuciosa provavelmente revelará que, na realidade, esses dados não compartilham os mesmos atributos.

**Dados geográficos que possuem os mesmos atributos, mas representam entidades diferentes devem ser separados em tabelas diferentes.**

Normalmente, em casos como este, uma análise mais minuciosa descobre diferenças de atributos entre as entidades. Mas nos casos raros e estranhos em que uma situação assim ocorrer é recomendado separar as entidades diferentes em tabelas diferentes.

## 3. Dados vetoriais no SDO\_GEOMETRY

O Oracle permite que os seguintes tipos de geometria para dados vetoriais sejam armazenados em um atributo *SDO\_GEOMETRY*: Ponto, Linha, Polígono, Multiponto, Multilinha, Multipolígono e Coleção.

De todos os tipos para dados vetoriais o ponto deve receber uma atenção especial quando for carregado ou instanciado dinamicamente em bloco anônimo, por exemplo. O *SDO\_GEOMETRY* possui o atributo *SDO\_POINT* que é específico para armazenar um ponto. Utilizando o *SDO\_POINT* para armazenar um ponto se interfere positivamente no desempenho das consultas espaciais que envolvem o objeto.

O Oracle possui vários construtores que preenchem o atributo *SDO\_POINT* automaticamente quando o dado é um ponto. Mas quando o ponto é construído manualmente deve se garantir que o atributo seja preenchido. Quando a carga do ponto é feita através de uma ferramenta não há como garantir o preenchimento do atributo, o que deve ser verificado e corrigido.

## 4. Metadados espaciais

Toda coluna do tipo *SDO\_GEOMETRY* precisa estar cadastrada no dicionário de metadados espaciais do Oracle. Os metadados espaciais registram o nome da tabela, o nome da coluna *SDO\_GEOMETRY*, o número de dimensões (eixos) do dado, os limites máximos para cada dimensão, a tolerância para cada dimensão e o sistema de referência espacial do dado.

As informações dos metadados espaciais são utilizadas para a validação do dado geográfico, para a indexação espacial e por vários operadores e funções espaciais. E por ser tão importante é essencial que o metadados seja muito bem entendido antes de ter seus valores definidos.

### Número de dimensões

É a quantidade de dimensões representadas na coluna espacial onde cada dimensão equivale a um eixo de coordenadas. Uma vez definido no metadados espacial, todos dados geográficos da tabela terão, obrigatoriamente, que ter todas suas coordenadas preenchidas para a quantidade de dimensões descrita no seu metadado.

### Limites de cada dimensão

São o limite inferior e o limite superior para os valores das coordenadas dos dados da camada espacial em uma dimensão específica. Os dados espaciais da tabela devem, obrigatoriamente, estar contidos dentro dos limites registrados no metadados espacial.

Estes limites não podem ser confundidos com o retângulo mínimo envolvente. Diferentemente do retângulo mínimo envolvente, os limites são paralelos aos eixos de coordenadas.

### Tolerância

É o valor que determina o nível de precisão do dado espacial. Define a distância limite entre dois objetos para que eles sejam considerados topologicamente separados. Este valor é utilizado em operações de validação e de busca topológica, entre outras, e é recomendado que seja escolhido de acordo com a precisão, ou resolução, utilizada na digitalização do dado espacial.

### Sistema de referência espacial

É o sistema de coordenadas no qual os dados espaciais estão referenciados. Pode ser geográfico – para coordenadas angulares; projetado – para coordenadas cartesianas; e também podem ser sistemas de coordenadas próprios – para sistemas de referência sem relação com a superfície da Terra e/ou específicos de algum aplicativo.

Todos os sistemas de coordenadas suportados pelo Oracle podem ser consultados na tabela *MDSYS.CS\_SRS*.

## Views e tabela de metadados espaciais

Os metadados espaciais ficam armazenados na tabela *SDO\_GEOM\_METADATA\_TABLE* do esquema *MDSYS*. Não é recomendado manipular os metadados geográficos diretamente nesta tabela.

Para manipulação dos metadados espaciais o Oracle disponibiliza a view atualizável USER\_SDO\_GEOM\_METADATA. Por esta view é possível consultar e manipular os metadados das tabelas espaciais específicas do usuário (*owner*) conectado ao banco de dados. Essa questão do usuário conectado acaba sendo uma limitação já que dessa forma o usuário não consegue ver nem registrar metadados de camadas armazenadas em outros usuários (*owners*).

Parte dessa limitação é resolvida pela view ALL\_SDO\_GEOM\_METADATA, que permite a um usuário consultar os metadados espaciais de todas as tabelas às quais tem acesso no banco de dados.

Mas não existem views para inserir, atualizar ou excluir metadados de outros usuários de banco. Para tanto, é necessário conectar ao banco com o usuário (*owner*) dono da camada espacial ou ter permissões sobre a tabela MDSYS.SDO\_GEOM\_METADATA\_TABLE, mas, como já foi dito, isso não é recomendado.

Os metadados geográficos do Oracle são informações para utilização específica do banco de dados sendo que não há a menor pretensão de ser um metadado geográfico completo para o usuário final, ainda que possuam algumas informações relevantes a ele.

## 5. Validação do formato do dado espacial

Os dados espaciais armazenados no banco de dados devem estar válidos para que as consultas funcionem de maneira correta.

### Funções de validação

O Oracle disponibiliza as seguintes funcionalidades para validação de dados espaciais:

#### **SDO\_GEOM.VALIDATE\_GEOMETRY\_WITH\_CONTEXT**

Procedure que valida uma única geometria. Valida os tipos de geometria e retorna a palavra 'TRUE' se geometria é válida ou um texto com informações do erro se a geometria é inválida. A validação verifica a representação da geometria do dado contra as definições de elementos estabelecidas pelo Open Gis Consortium – OGC<sup>1</sup> no documento de especificação de padrões Simple Feature Specification for SQL.

Esta procedure possui duas assinaturas, uma com o valor de tolerância como parâmetro e outra com o tipo SDO\_DIM\_ARRAY com as informações de dimensões, limites das dimensões e a tolerância para cada dimensão.

#### **SDO\_GEOM.VALIDATE\_LAYER\_WITH\_CONTEXT**

Mesma coisa da VALIDATE\_GEOMETRY\_WITH\_CONTEXT, mas a verificação ocorre para todos os dados espaciais de uma coluna SDO\_GEOMETRY de uma tabela e os resultados são inseridos em uma outra tabela que deve ser criada pelo próprio usuário.

Esta função não recebe parâmetros pois utiliza as informações dos metadados espaciais da tabela e da coluna espacial para fazer as validações.

#### **SDO\_GEOMETRY.ST\_ISVALID**

Esta função é, na realidade, um método do objeto SDO\_GEOMETRY. Faz a mesma verificação da VALIDATE\_GEOMETRY\_WITH\_CONTEXT, mas seu retorno é diferente sendo 1 (um) se a geometria for

---

<sup>1</sup> <http://www.opengeospatial.org/>



válida e 0 (zero) se for inválida. Outra diferença é que este método não recebe parâmetros e utiliza sempre 0.001 como tolerância.

## Critérios de validação

Quando as informações das dimensões ou as informações dos metadados são passadas como parâmetro as funções `VALIDATE_GEOMETRY_WITH_CONTEXT` ou `VALIDATE_LAYER_WITH_CONTEXT` validam se objeto está dentro dos limites das dimensões definidos no `SDO_DIM_ARRAY`.

A tolerância é utilizada para validar se os vértices da geometria não são redundantes, pois os pontos que definem os vértices devem ser distintos e, portanto, devem obedecer a distância mínima entre eles.

Uma linha deve ter dois ou mais pontos.

Os vértices inicial e final do polígono devem ser os mesmos.

Um polígono deve ter no mínimo quatro vértices.

Os lados de um polígono não podem se interceptar.

A orientação dos vértices do anel externo de um polígono de duas dimensões deve ser anti-horária e a orientação do anel interno, quando existir, deve ser horária.

## Erros mais comuns em geometrias inválidas

Os erros de validação mais comuns para dados vetoriais estão listados na **Tabela 1**. O código do erro é aquele retornado nos cinco primeiros caracteres do texto retornado pelas as funções `VALIDATE_GEOMETRY_WITH_CONTEXT` ou `VALIDATE_LAYER_WITH_CONTEXT`.

Código do erro	Descrição	Possíveis causas	Possíveis soluções
13356	Vértices adjacentes redundantes na geometria.	A tolerância pode ter sido mal atribuída.	Ajustar a tolerância para o valor real da resolução do dado.
		O dado foi desenhado com os vértices mais próximos do que deveriam.	Corrigir o desenho.
13349	Limite do polígono intercepta a ele mesmo.	A tolerância pode ter sido mal atribuída.	Ajustar a tolerância para o valor real da resolução do dado.
		O polígono foi desenhado errado e os limites se cruzam.	Corrigir o desenho.
13367	Sentido incorreto dos vértices no polígono externo ou no interno.	O polígono foi desenhado errado.	Corrigir o desenho do polígono. Vértices de anéis externos devem estar no sentido anti-horário e vértices de anéis internos no sentido horário.

<b>13343</b>	A geometria do polígono possui menos de quatro vértices.	O polígono foi desenhado errado.	Corrigir o desenho do polígono. Um polígono deve possuir no mínimo quatro vértices sendo que, para indicar o fechamento do polígono, o vértice inicial e o final devem ter a mesmas coordenadas.
<b>13351</b>	Dois ou mais polígonos de uma geometria complexa se interceptam.	A tolerância pode ter sido mal atribuída.  Os polígonos que compõem o objeto complexo foram desenhados errados e os limites deles se cruzam.	Ajustar a tolerância para o valor real da resolução do dado.  Corrigir o desenho do polígono.

Tabela 1: Erros mais comuns em geometrias inválidas.

## Funções para correção de geometrias inválidas

O Oracle disponibiliza mecanismos para correção de alguns dos erros encontrados. Elas são importantes para tratar erros mais simples e para rastrear erros mais complexos. São elas:

### **SDO\_UTIL.REMOVE\_DUPLICATE\_VERTICES**

Remove vértices redundantes em uma geometria. Recebe um objeto SDO\_GEOMETRY e a tolerância como parâmetros de entrada e retorna um objeto SDO\_GEOMETRY sem os vértices redundantes.

### **SDO\_UTIL.RECTIFY\_GEOMETRY**

Conserta uma geometria inválida nos casos de vértices redundantes, de limite de polígono interceptando a ele mesmo e de sentido incorreto nos vértices de um polígono. Recebe um objeto SDO\_GEOMETRY e a tolerância como parâmetros de entrada e retorna um objeto SDO\_GEOMETRY.

### **SDO\_UTIL.EXTRACT**

Extrai um elemento de uma geometria complexa. É muito útil para inspecionar uma geometria e rastrear erros na mesma. Um bom exemplo é extrair um polígono de um multipolígono. Recebe um objeto SDO\_GEOMETRY, um número que identifica o elemento e a ser extraído e, opcionalmente, um identificador do anel (interno ou externo) do polígono para os casos de polígono com buraco. Ela retorna um objeto SDO\_GEOMETRY do elemento extraído.

### **SDO\_UTIL.GETNUMELEM**

Retorna o número de elementos de uma geometria. Também é útil para inspecionar uma geometria e rastrear erros na mesma. Recebe um objeto SDO\_GEOMETRY como parâmetro de entrada e retorna o número de elementos da geometria.

### **SDO\_UTIL.GETNUMVERTICES e SDO\_UTIL.GETVERTICES**

Permitem inspecionar os vértices de uma geometria. A primeira retorna o número de vértices de uma geometria e recebe um objeto SDO\_GEOMETRY como parâmetro de entrada. A segunda retorna as coordenadas dos vértices em objeto VERTEX\_SET\_TYPE (que nada mais é que uma tabela de vértices) e

recebe um objeto SDO\_GEOMETRY como entrada. São funções úteis para rastrear vértices que estão fazendo uma geometria ser inválida.

## 6. Indexação espacial

A indexação espacial é essencial para o desempenho das consultas espaciais sendo obrigatória sua existência para a utilização de alguns operadores espaciais (isto será visto mais adiante na seção 7). Quando bem configurados, os índices também podem ajudar na manutenção da integridade dos dados e na melhor utilização do espaço ocupado pelo índice.

Os índices espaciais são implementados como árvores R. A hierarquização da árvore é feita por retângulos aproximados das geometrias dos objetos de um atributo SDO\_GEOMETRY de uma tabela. Por exemplo, a raiz da árvore será o retângulo mínimo envolvente de todos os dados de uma coluna SDO\_GEOMETRY de uma tabela e nos níveis abaixo da raiz existirão retângulos mínimos envolventes de agrupamentos distintos dos dados sendo que os agrupamentos diminuem à medida que o número de níveis aumenta até as folhas da árvore R.

### Parâmetros

O comando de criação de índices possibilita a atribuição de alguns parâmetros importantes para o melhor funcionamento dos índices espaciais. São eles:

#### **tablespace**

Determina a tablespace onde o índice será armazenado. Caso este parâmetro não seja especificado o índice será criado na tablespace do owner dele.

#### **work\_tablespace**

Determina a tablespace de trabalho a ser utilizada pelo Oracle durante a criação do índice.

Durante a criação do índice espacial são feitas operações de ordenação de todo conjunto de dados espaciais da tabela e essas operações criam algumas tabelas de trabalho que são automaticamente eliminadas após a criação do índice. Criar e eliminar várias tabelas de vários tamanhos acaba por fragmentar o espaço utilizado na tablespace. Então, a utilização de uma tablespace separada para as tabelas de trabalho evita essa fragmentação.

A tablespace de trabalho pode ser eliminada após a criação do índice e deve ter duas vezes o tamanho final do índice.

Caso este parâmetro não seja especificado as tabelas de trabalho serão criadas na tablespace do índice espacial.

#### **layer\_gtype**

Determina o tipo de geometria do atributo espacial de uma tabela. Atua como uma restrição, pois após a criação do índice apenas o tipo de geometria determinado no parâmetro será permitido no atributo espacial.

Seu valor pode ser point (ponto), line (linha), polygon (polígono), collection (coleção), multipoint (multiponto), multiline (multilinha) e multipolygon (multipolígono).

A especificação do tipo de geometria também melhora o desempenho das consultas espaciais. Pois faz com que otimizações internas sejam invocadas durante as consultas, especialmente se o tipo da geometria indexada for ponto.

Caso este parâmetro não seja especificado o tipo de geometria padrão adotado para o índice é coleção.

### **sdo\_indx\_dims**

Determina a quantidade de dimensões do dado espacial indexado e caso não seja especificado a quantidade de dimensões padrão, que são duas, é atribuída ao índice.

O **Exemplo 1** mostra um comando de criação do índice espacial `sx_quadra` para a coluna geometria da tabela `quadra` onde os dados são polígonos de duas dimensões, a `tablespace` para o índice espacial é a `sx_tabspce` e a `tablespace` de trabalho é `wk_tabspce`.

**Exemplo 1: Comando de criação de índice espacial.**

```
SQL> CREATE INDEX SX_QUADRA ON QUADRA(GEOMETRIA)
      INDEXTYPE IS MDSYS.SPATIAL_INDEX
      PARAMETERS ('tablespace=sx_tabspce
                  work_tablespace=wk_tabspce
                  layer_gtype=polygon
                  sdo_indx_dims=2');
```

As views `USER_SDO_INDEX_METADATA` e `ALL_SDO_INDEX_METADATA` armazenam informações sobre os índices espaciais do Oracle, incluindo os parâmetros especificados na criação deles.

## **7. Análise espacial**

O Oracle disponibiliza operadores e funções para análise espacial.

### **Operadores espaciais**

São operadores que verificam relacionamentos espaciais entre dois objetos espaciais. São utilizados na cláusula `where` de comando SQL e obedecem a uma sintaxe padrão apresentada no **Exemplo 2**.

**Exemplo 2: Sintaxe padrão dos operadores espacial**

```
<OPERADOR ESPACIAL>(coluna espacial da tabela pesquisada,
                    objeto espacial da consulta,
                    [parâmetros do operador espacial]) = 'TRUE'
```

O primeiro argumento é sempre a coluna espacial da tabela pesquisada, o segundo é sempre o objeto espacial da consulta e o terceiro são parâmetros específicos para alguns operadores.

A coluna espacial da tabela pesquisada deve, obrigatoriamente, estar indexada. Caso contrário o operador retorna um erro.

O operador espacial deve sempre ser comparado com a palavra `'TRUE'` e sempre através da verificação de igualdade. Quando quiser um resultado diferente do que um determinado operador verifica o recomendado é utilizar operadores diferentes.

Os operadores utilizam a tolerância definida nos metadados espaciais.

Sua utilização é análoga à utilização de operadores relacionais.

Exemplos de operadores espaciais são SDO\_RELATE, SDO\_FILTER, SDO\_WITHIN\_DISTANCE, SDO\_NN, etc.

### **ORDERED hint**

Quando o segundo argumento do operador espacial são múltiplos objetos que vem de uma tabela é recomendada a utilização da *hint* ordered. Esta *hint* não é específica para consultas espaciais, com ela o otimizador do Oracle altera, internamente, o plano de execução para melhorar o desempenho da consulta. Quando se utiliza a *hint* ordered a tabela que alimenta o segundo argumento do operador deve estar listada na cláusula where antes da tabela que alimenta o primeiro operador.

### **Funções espaciais**

As funções espaciais também fazem análise espacial, mas são um pouco diferentes dos operadores. A indexação espacial não é obrigatória para elas sua aplicação não é restrita à cláusula where podendo ser utilizados, também, na select.

Exemplos de funções espaciais são SDO\_GEOM.SDO\_DISTANCE, SDO\_GEOM.RELATE, SDO\_GEOM.SDO\_AREA, SDO\_GEOM.SDO\_LENGTH, etc.

## **8. Recomendações finais**

Mantenha o *patch set* do Oracle atualizado, sempre que possível.

Planeje bem seu banco de dados. A maioria dos problemas, inclusive de desempenho, pode ser evitada quando o esforço adequado for dedicado à análise e modelagem dos requisitos de dados.

Não existem restrições para que uma tabela possua mais de um atributo espacial, isso depende unicamente do negócio envolvendo o dado. O que deve ser feito é seguir as recomendações de modelagem para cada um dos atributos espaciais.

Caso os dados espaciais já estejam validados antes de serem carregados no banco de dados não há necessidade de validação deles no banco. Caso contrário a validação é altamente recomendada.

Os pacotes SDO\_GEOM e SDO\_UTIL possuem várias outras funções úteis para inspeção e tratamento do dado espacial.

Monitore o banco em busca de gargalos de desempenho, analise-os e solucione-os.

Analise o plano de execução em busca de falhas e melhorias em suas consultas espaciais.

Consulte o Apêndice B do Oracle Spatial Developers Guide <sup>2</sup> para saber quais recursos são licenciados para o Locator e quais não são.

---

<sup>2</sup> [http://docs.oracle.com/cd/E11882\\_01/appdev.112/e11830/sdo\\_locator.htm#i632018](http://docs.oracle.com/cd/E11882_01/appdev.112/e11830/sdo_locator.htm#i632018)