SQLServerFast.com Execution Plan Video Training

Block 2: Reading data

Level: Advanced

Chapter 3: Special index types

Filtered indexes

```
CREATE INDEX ix_Person_LastName_filtered
ON Person.Person (LastName)
WHERE is_deleted = 0;
```

- Filtered indexes
  - Nonclustered on-disk rowstore index
  - Nonclustered on-disk columnstore index
  - Only rows that meet the predicate are included in the index
    - Index only used when all needed rows are guaranteed to be there
    - Smaller index, so less I/O needed
    - Storage structure not affected
    - Same Index Scan and Index Seek operators

Filtered indexes

```sql
CREATE INDEX ix_Person_LastName_filtered
ON Person.Person (LastName)
WHERE is_deleted = 0;
```

- Filtered indexes
  - Optimizer must ensure results are correct
    - Query predicate *might* include rows not in the filtered index?
      - Filtered index cannot be used!
      - Error when you try to force it
    - Query predicate is a *subset* of the filtered index?
      - Lookup might be needed to remove unwanted rows
        - Sometimes even done when the query predicate IS an exact match of the index filter
    - Best practice: INCLUDE all filter columns

XML indexes

- XML indexes
  - Used for columns with data type xml
    - Stored internally in (undocumented) shredded format
      - Reconstructed to XML form when returned to client
        - This might affect formatting and whitespace
        - Store XML document as nvarchar(max) is exact formatting is relevant!

XML indexes

- XML indexes
  - Used for columns with data type xml
    - Stored internally in (undocumented) shredded format
      - Reconstructed to XML form when returned to client
      - Not efficient when filtering on the contents of the XML column
        - Would need to reconstruct each XML value before testing the predicate
        - XML indexes help

XML indexes

- XML indexes
  - Used for columns with data type xml
  - Four types
    - Primary XML index
    - Secondary XML index
      - FOR PATH
      - FOR PROPERTY
      - FOR VALUE
    - “Selective” XML index
      - Specifies which part(s) of the XML are included
      - Similar to filtered indexes

XML indexes

- XML indexes
  - Used for columns with data type xml
  - Four types
    - Primary XML index
      - Internal table (“node table”) with clustered index
      - Two or more rows for each leaf node
    - Secondary XML index
      - Nonclustered index on the internal node table
        - FOR PATH → on encoded representation of the path
        - FOR VALUE → on the value
        - FOR PROPERTY → on row’s clustered index + path + value

XML indexes

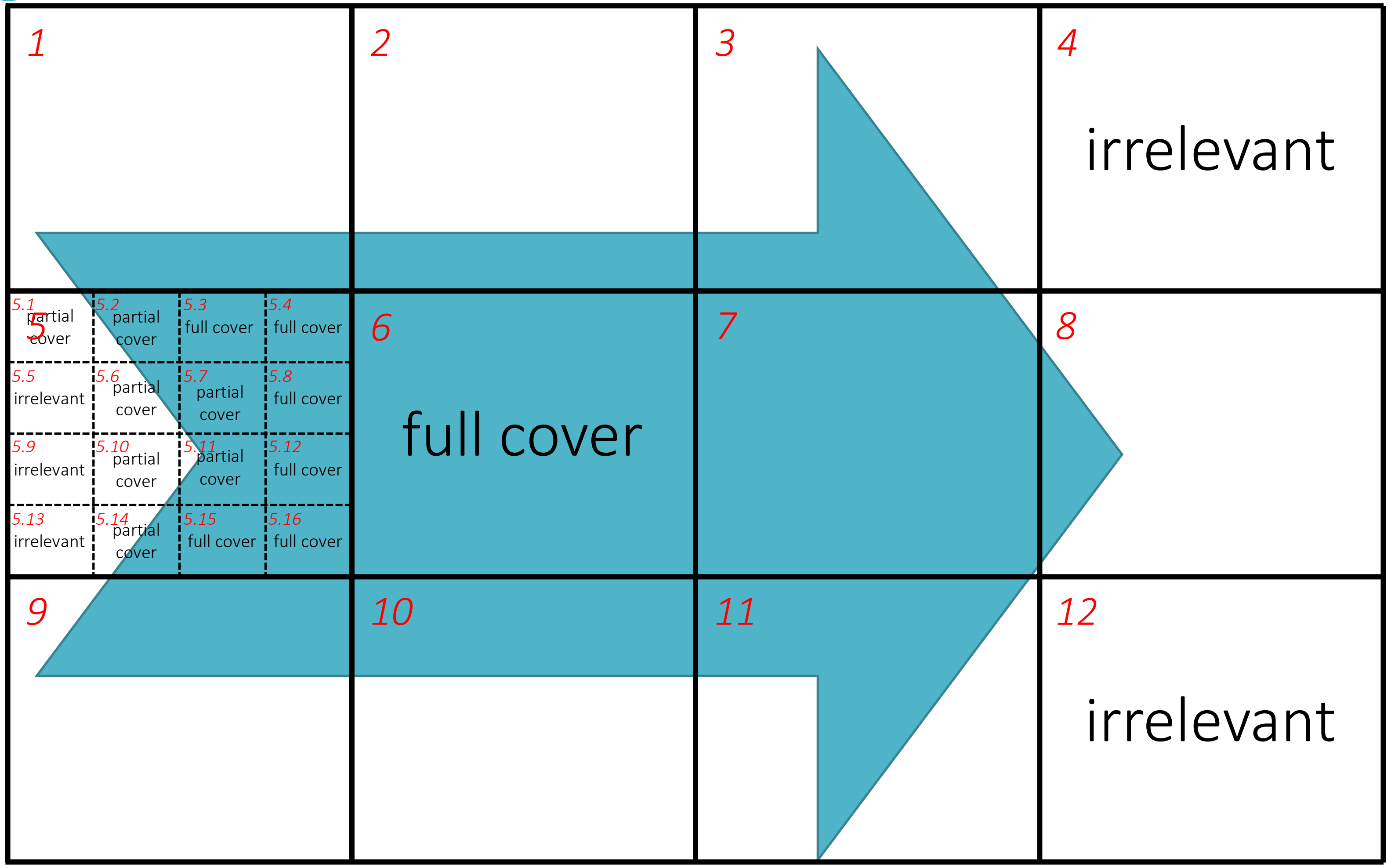
- XML indexes
  - Used for columns with data type xml
  - Four types
    - Primary XML index
      - Clustered index on internal structure
    - Secondary XML index
      - Nonclustered index on internal structure
    - Read with normal operators
      - Index Scan to read all rows
      - Index Seek to selectively access specific rows

Spatial indexes

Spatial indexes

Tessellation

Cell
6
5.3
5.4
5.8
5.12
5.15
5.16
5.1
5.2
5.6
5.7
5.10
5.11

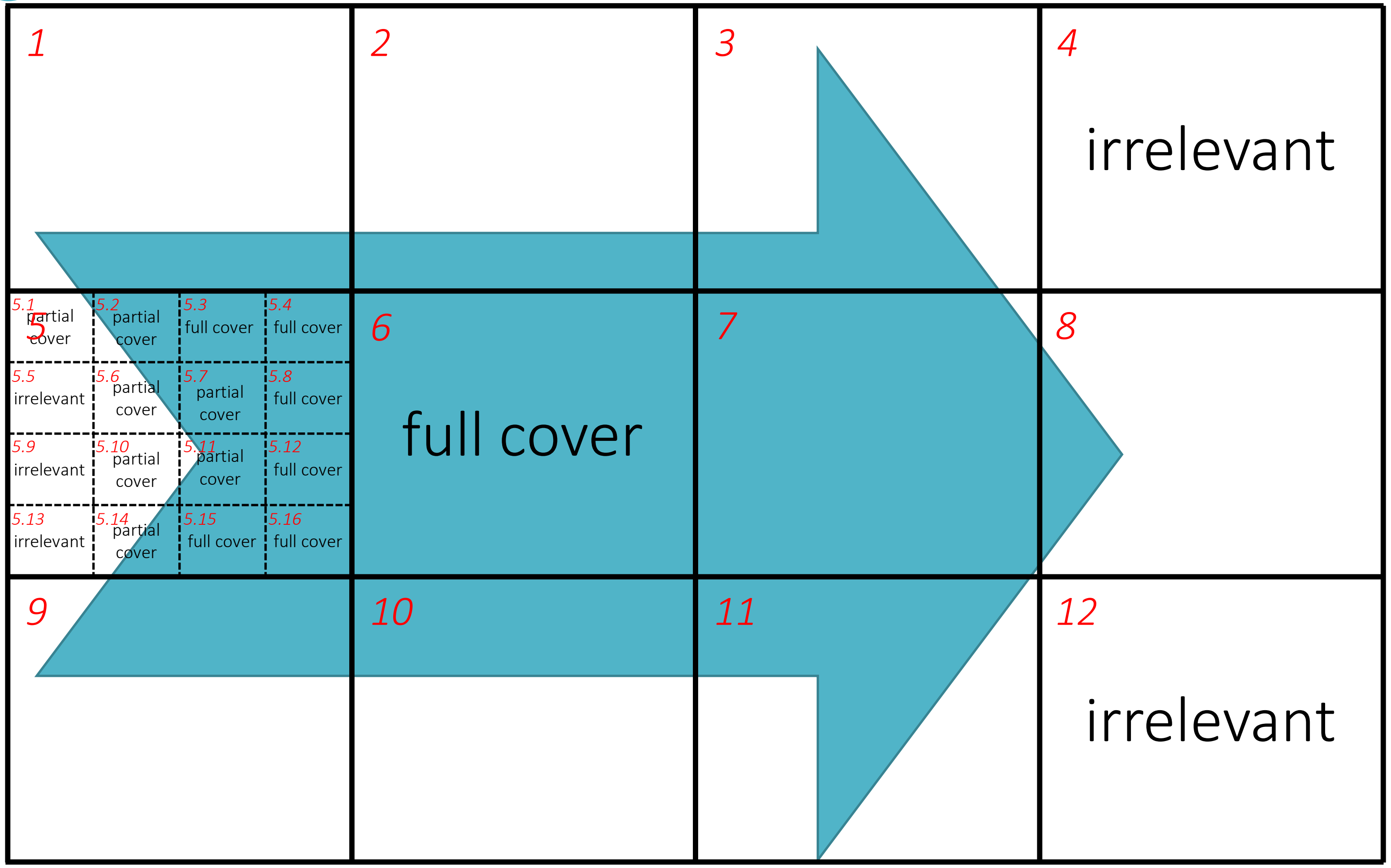


Spatial indexes

Spatial indexes

Tessellation

Cell	Coverage
6	Full
5.3	Full
5.4	Full
5.8	Full
5.12	Full
5.15	Full
5.16	Full
5.1	
5.2	
5.6	
5.7	
5.10	
5.11	

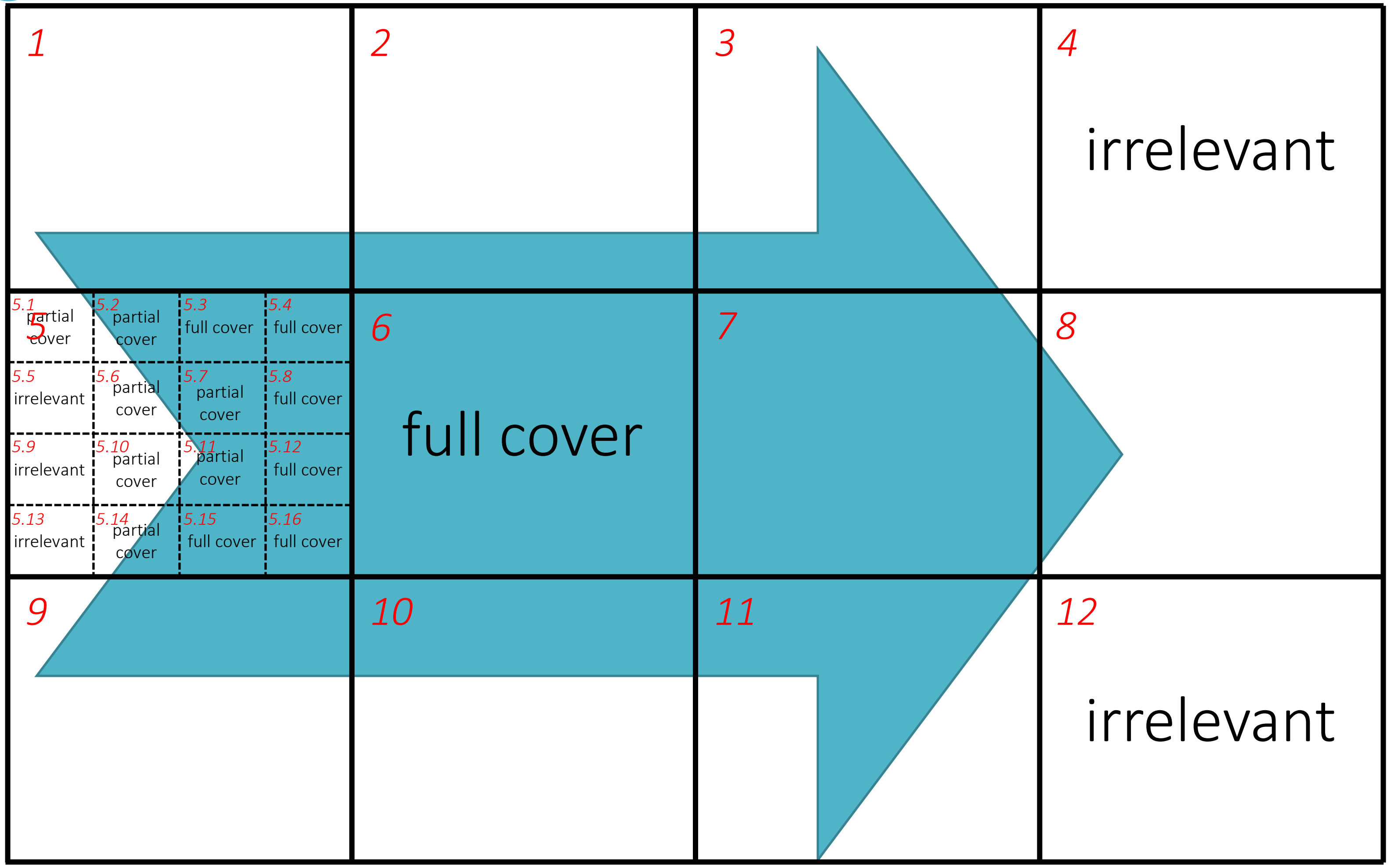


Spatial indexes

Spatial indexes

Tessellation

Cell	Coverage
6	Full
5.3	Full
5.4	Full
5.8	Full
5.12	Full
5.15	Full
5.16	Full
5.1	Partial
5.2	Partial
5.6	Partial
5.7	Partial
5.10	Partial
5.11	Partial

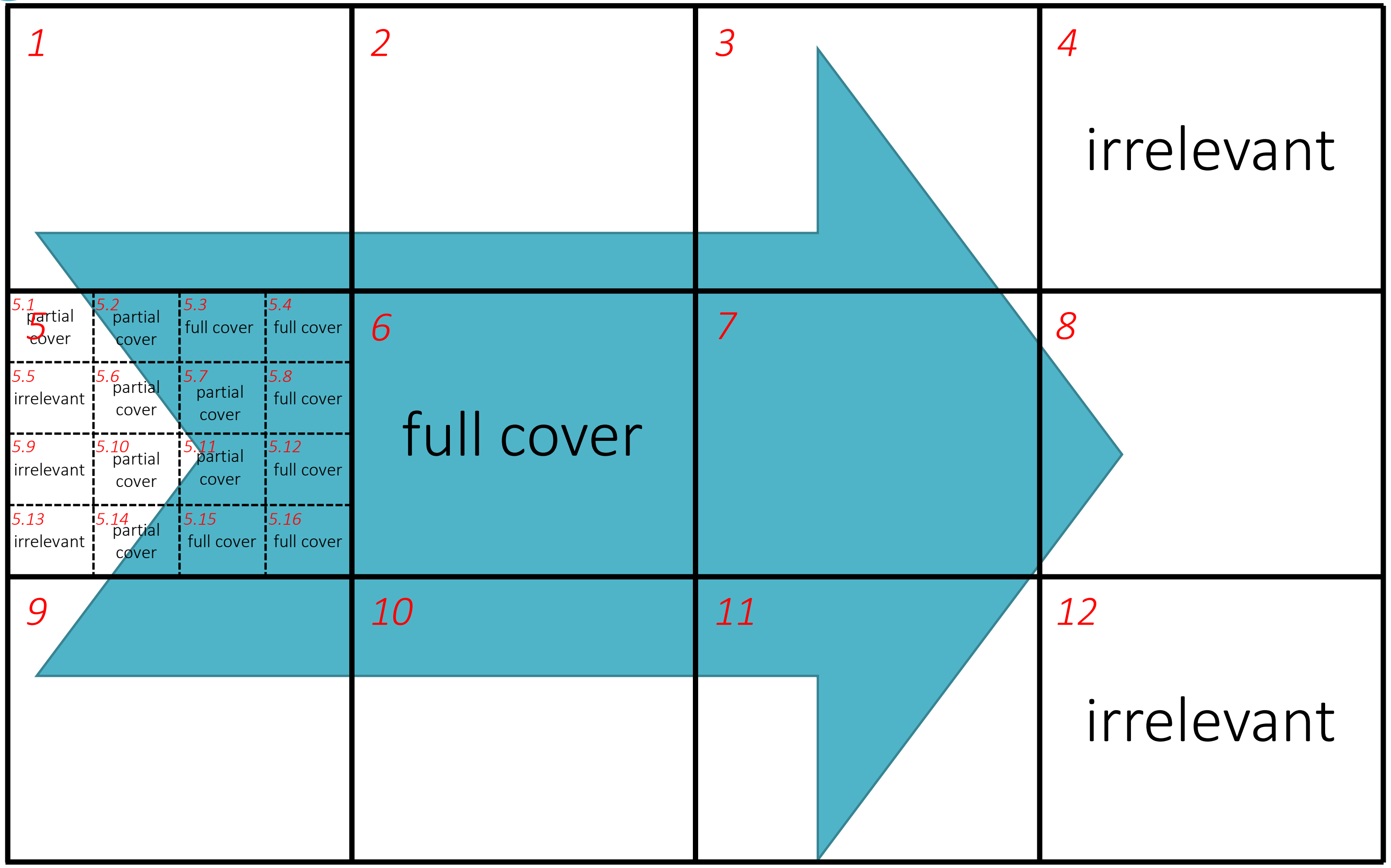


Spatial indexes

Spatial indexes

Tessellation

Cell	Coverage	Cl.Ind.Key
6	Full	3865
5.3	Full	3865
5.4	Full	3865
5.8	Full	3865
5.12	Full	3865
5.15	Full	3865
5.16	Full	3865
5.1	Partial	3865
5.2	Partial	3865
5.6	Partial	3865
5.7	Partial	3865
5.10	Partial	3865
5.11	Partial	3865

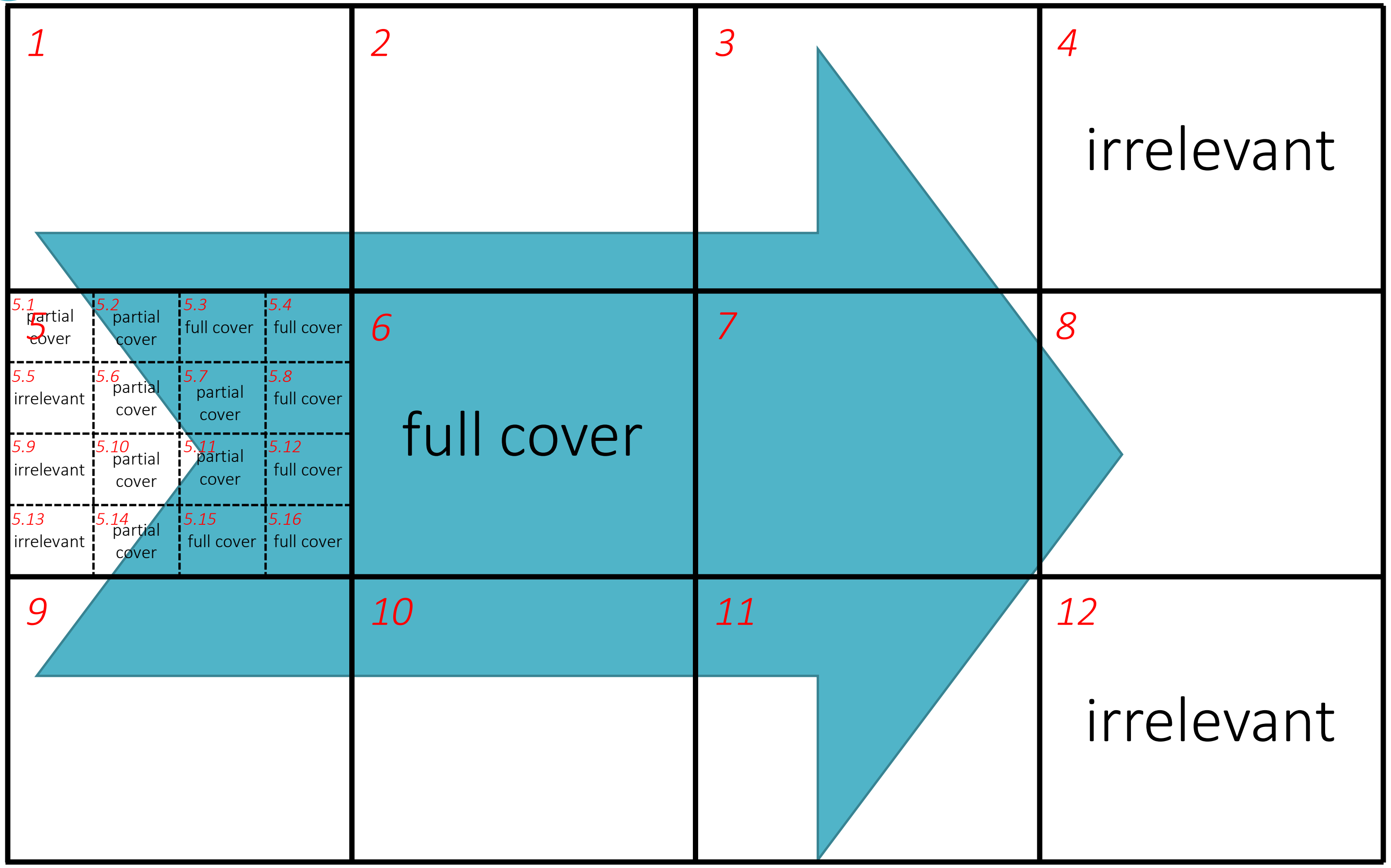


Spatial indexes

Spatial indexes

Tessellation

Cell	Attr	Cl.Ind.Key
6	2	3865
5.3	2	3865
5.4	2	3865
5.8	2	3865
5.12	2	3865
5.15	2	3865
5.16	2	3865
5.1	1	3865
5.2	1	3865
5.6	1	3865
5.7	1	3865
5.10	1	3865
5.11	1	3865

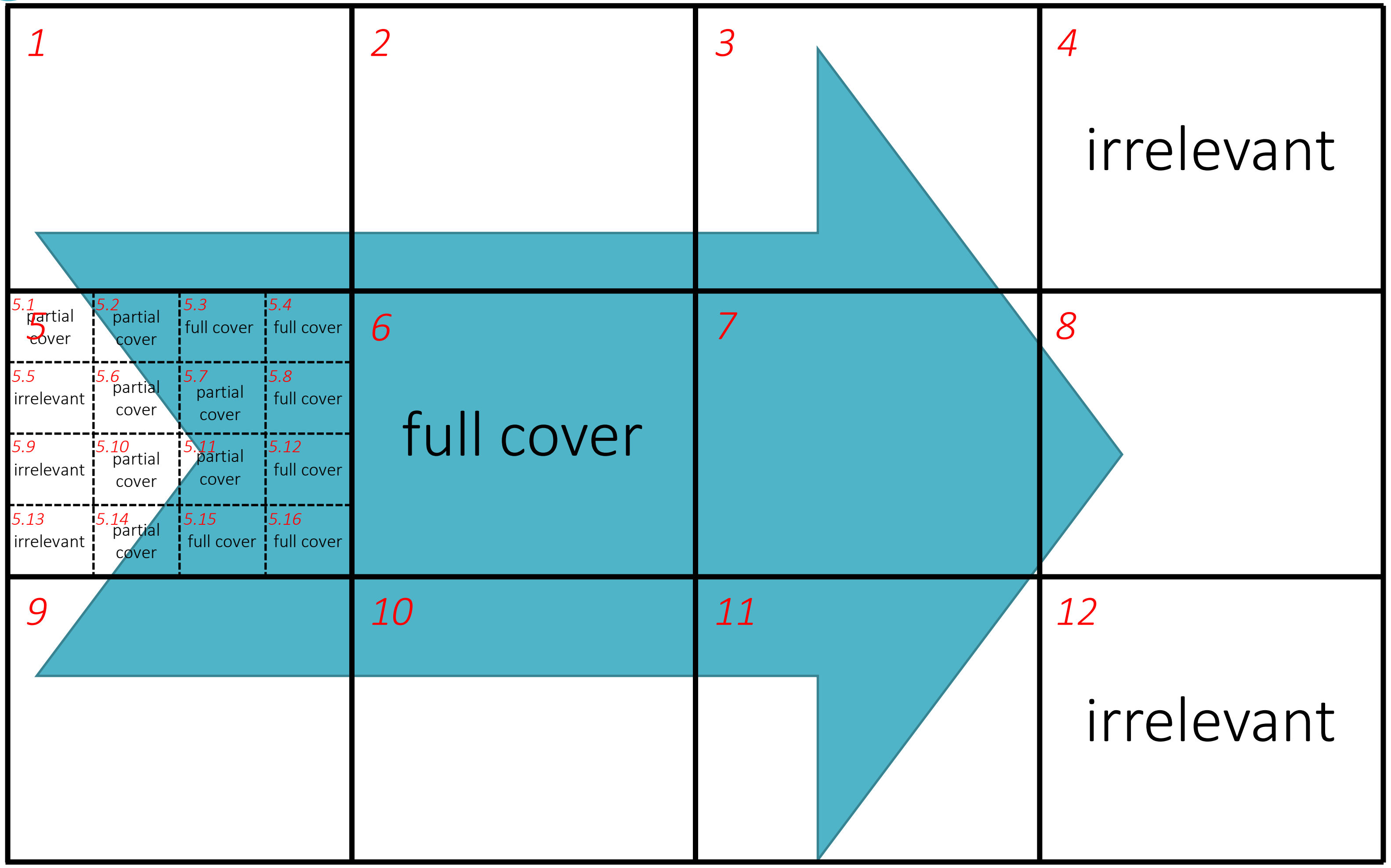


Spatial indexes

Spatial indexes

Tessellation

Id	Attr	Cl.Ind.Key
6	2	3865
5.3	2	3865
5.4	2	3865
5.8	2	3865
5.12	2	3865
5.15	2	3865
5.16	2	3865
5.1	1	3865
5.2	1	3865
5.6	1	3865
5.7	1	3865
5.10	1	3865
5.11	1	3865

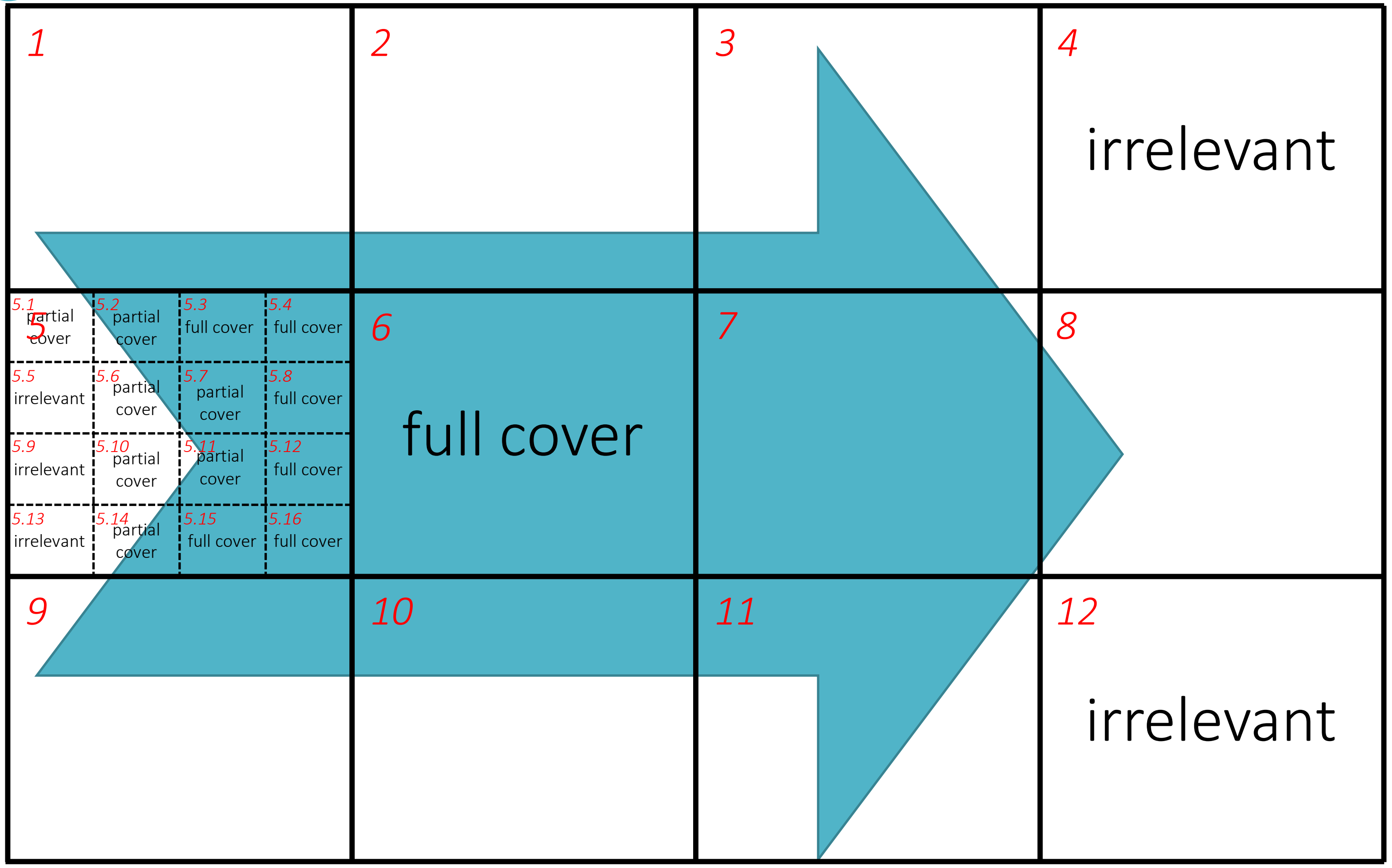


Spatial indexes

Spatial indexes

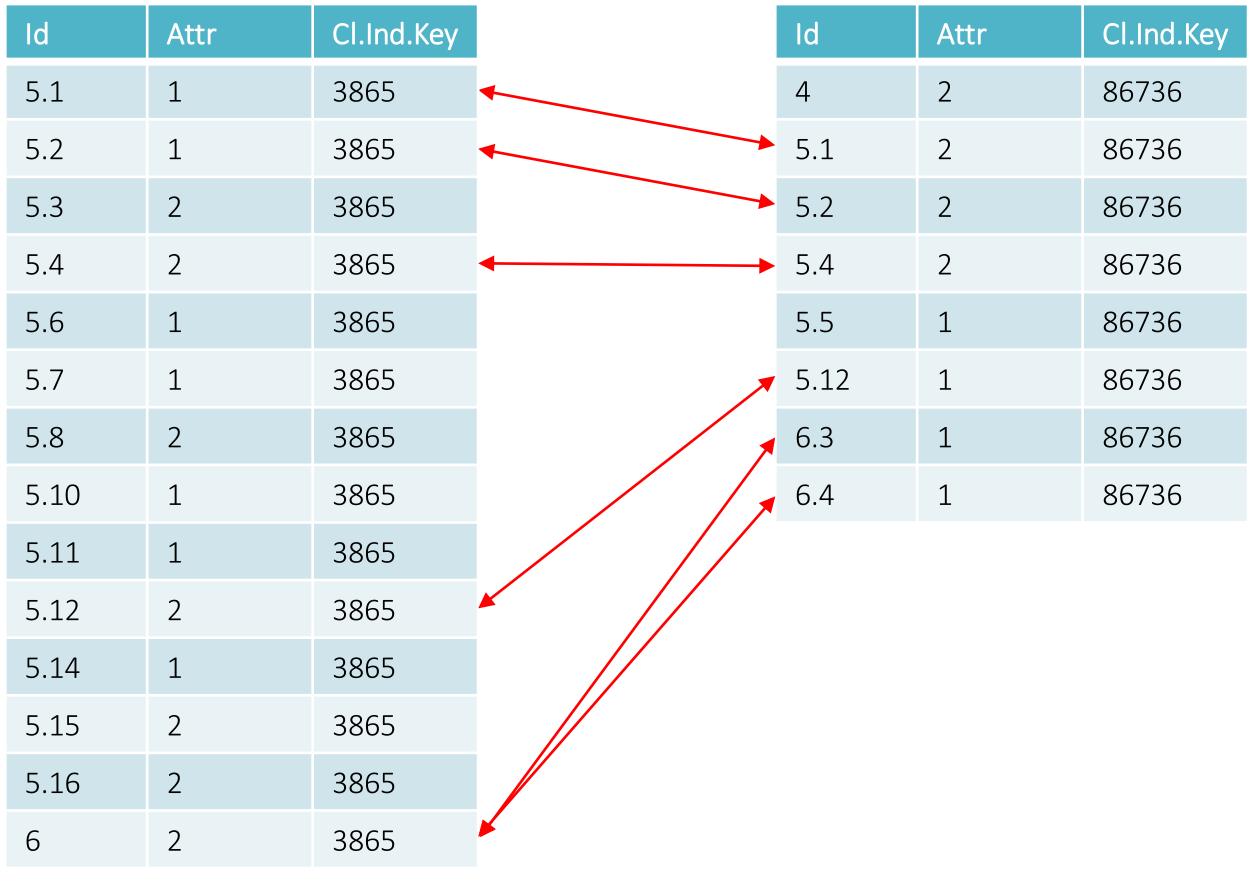
Tessellation

Id	Attr	Cl.Ind.Key
5.1	1	3865
5.2	1	3865
5.3	2	3865
5.4	2	3865
5.6	1	3865
5.7	1	3865
5.8	2	3865
5.10	1	3865
5.11	1	3865
5.12	2	3865
5.14	1	3865
5.15	2	3865
5.16	2	3865



Spatial indexes

Spatial indexes
- Tessellation
  - STIntersects()



Id	Attr	Cl.Ind.Key
5.1	1	3865
5.2	1	3865
5.3	2	3865
5.4	2	3865
5.6	1	3865
5.7	1	3865
5.8	2	3865
5.10	1	3865
5.11	1	3865
5.12	2	3865
5.14	1	3865
5.15	2	3865
5.16	2	3865
6	2	3865

Id	Attr	Cl.Ind.Key
4	2	86736
5.1	2	86736
5.2	2	86736
5.4	2	86736
5.5	1	86736
5.12	1	86736
6.3	1	86736
6.4	1	86736

Spatial indexes

- Spatial indexes
  - Tessellation
    - STIntersects()
      - No logic needed for rows with no overlapping cells
      - Cheap test only for rows with overlapping cells with full coverage
      - Expensive test only for rows with overlapping partially covered cells

Spatial indexes

- Spatial indexes
  - Tessellation
  - Standard clustered index
  - Additional operators for tessellation and partial/full coverage handling
    - These are different depending on methods used

Full-text indexes

- Full-text indexes
  - Originally a third party product
  - Now integrated in SQL Server
    - But internals were never *fully* integrated!
    - Stored as blob pages, using proprietary format
    - Accessed by special components
      - Accessed through Table Valued Function operator
      - Internals undocumented / unknown
      - Returns clustered index key value of relevant rows

Summary

- Special index types
  - Filtered indexes
  - XML Indexes
  - Spatial indexes
  - Full-text indexes

Summary

- Special index types
  - Filtered indexes
    - Same structure, less data

Summary

- Special index types
  - Filtered indexes
  - XML Indexes
    - Clustered and nonclustered indexes on special structures
    - Standard scan and seek operators
  - Spatial indexes
    - Clustered indexes on special structures
    - Standard scan and seek operators

Summary

- Special index types
  - Filtered indexes
  - XML Indexes
  - Spatial indexes
  - Full-text indexes
    - Undocumented proprietary format
    - Accessed in execution plan through Table Valued Function operator

Next chapters

Chapter 4: Reading data in parallel or batch mode

- Parallel page supplier
- Batch mode scans on rowstore and columnstore indexes

Chapter 5: Assorted read optimizations