<epam>

Normal Forms – Part 1

Relational Databases Basics

TRAINING
CENTER

<epam>

Disclaimer

Usually most people understand 1NF-3NF easily.
Speaking about BCNF and further forms:
~ 10% understand them at once.
~ 80% understand them after 1-2 years of experience.
~ 10% never understand them.

I hope you are here! ☺

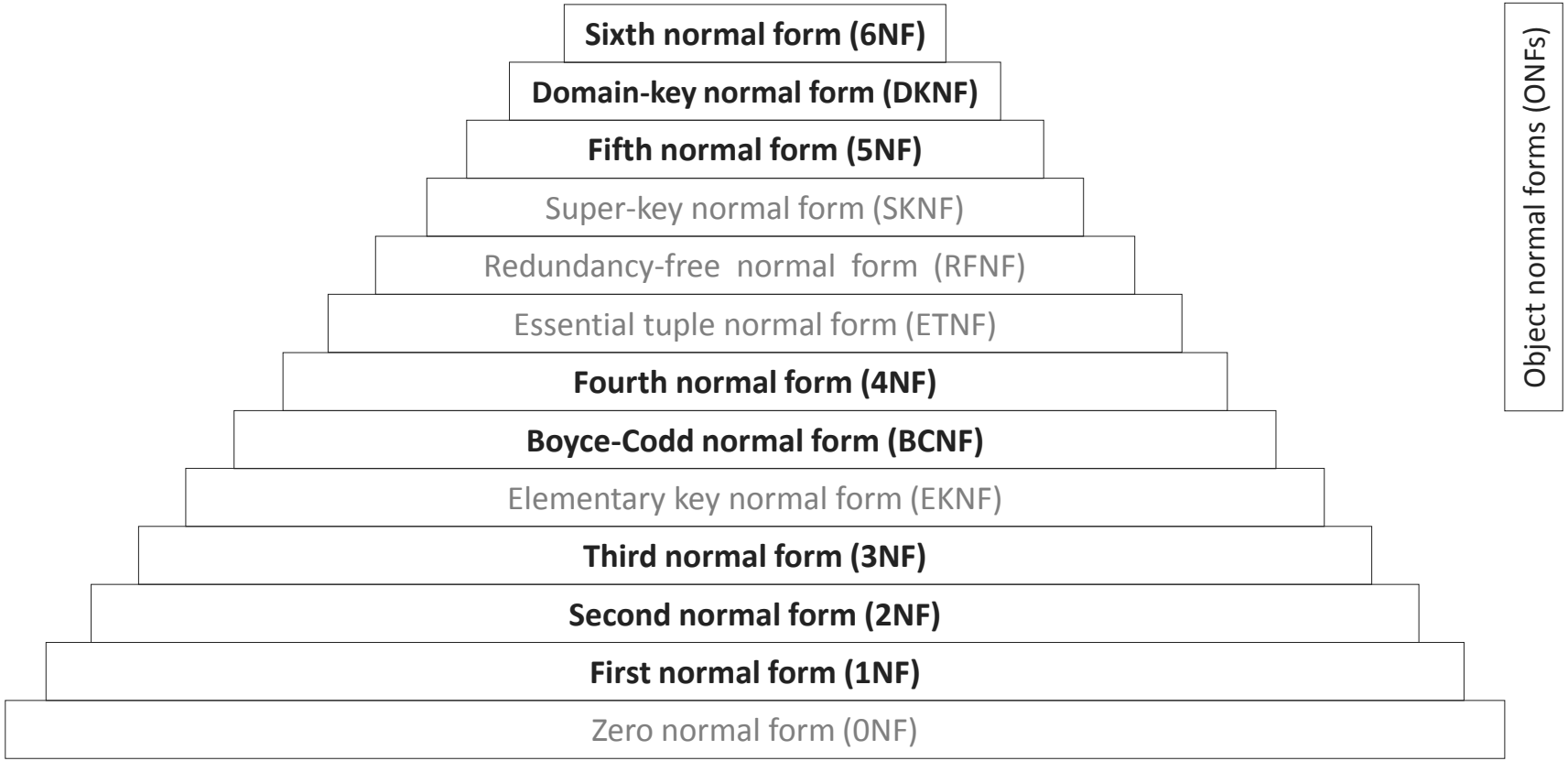
Any relation variable may (or may not) be in any normal form: it depends on subject matter ONLY. I.e. if subject matter rules change, any normal form may (or may not) be violated for any relation.

Read and remember!

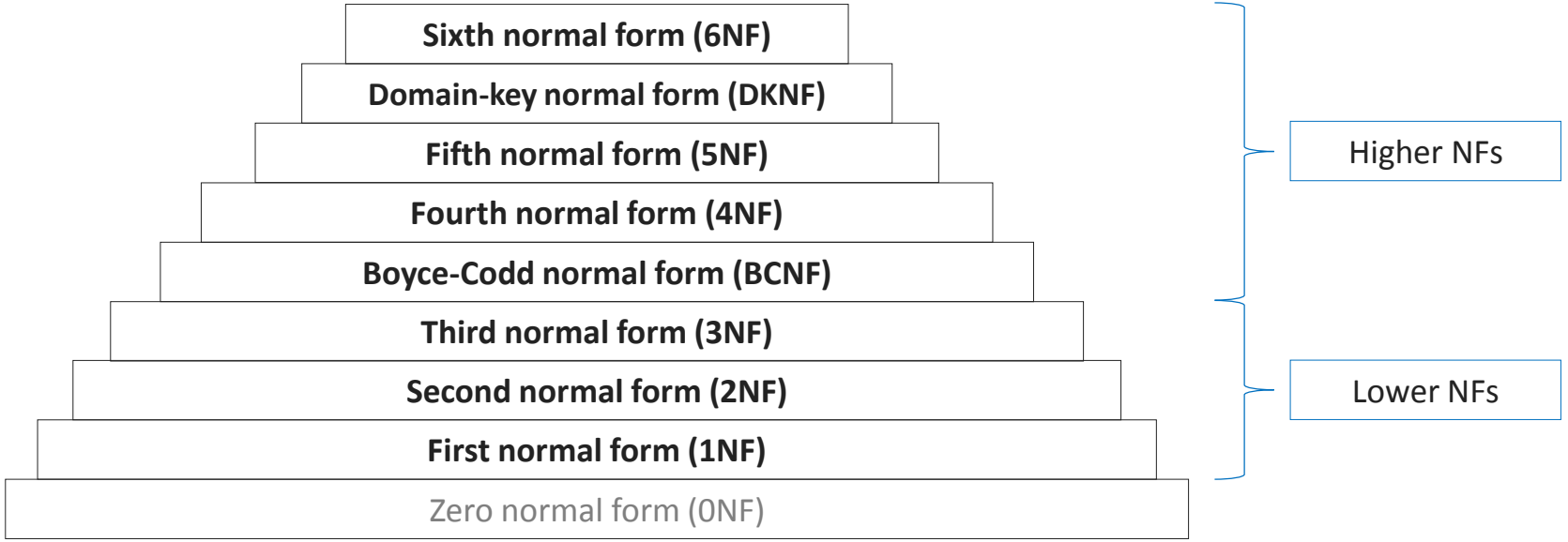
Normalization – a process of decomposition of relation variable R into projections R_1, R2, ..., R_n, such that:

- the join of R_1, R_2, ..., R_n is guaranteed to be equal to R;
- each of R_1, R_2, ..., R_n is needed in order to provide that guarantee;
- at least one of R_1, R_2, ..., R_n is at a higher level of normalization than R is.

Normal forms hierarchy



Normal forms hierarchy



0NF: zero normal form

Read and remember!

A relation variable (relvar) is in **0NF** if it violates any higher NFs requirements, i.e. if it “is not in any NF”.

0NF is “non-canonical” one, but it is a good start point for any further discussions.

1NF: first normal form

Read and remember!

A relvar is in **1NF** if and only if every tuple contains exactly one value for each attribute.

In other words: every value must be an atomic one, i.e. must NOT be multi-valued or compound one.

1NF: first normal form (violation examples)

student

s_id (PK)	s_mark
1731	{3, 6, 8, 6, 4, 6, 7}
1824	…

Multi-valued attribute

student

s_id (PK)	s_address
1731	Big City, New Street, building 1
1824	…

Compound attribute

student

s_id (PK)	s_result
1731	subject: Chemistry, mark: 8; subject: Physics, mark: 9
1824	…

Combination of “multi-valued attributes” and “compound attributes” (actually, “a table within a table”)

1NF: first normal form (dealing with violations)

Before

student

s_id (PK)	s_mark
1731	{3, 6, 8, 6, 4, 6, 7}
1824	…

Multi-valued attribute

After

student

s_id (PK)	…
1731	…

1-M relationship

mark

m_id (PK)	m_student (FK)	m_mark
1123	1731	3
1124	1731	6

1NF: first normal form (dealing with violations)

Before

student

s_id (PK)	s_address
1731	Big City, New Street, building 1
1824	…

Compound attribute

After

student

s_id (PK)	s_addr_city	s_addr_street	s_addr_building
1731	Big City	New Street	1
1824	…	…	…

1NF: first normal form (atomicity depends on subject matter)

Shall we store in separate fields…

- The phone number and the operator code (prefix)?
- The e-mail parts (username and domain)?
- The date parts (year, month, day)?
- The time parts (hours, minutes, seconds, milliseconds)?
- The document id parts (like passport series id and number)?

1NF: first normal form (conclusion)

- ✓ Most DBMSes will not allow you to store “a table within a table”.
- ✓ Each multi-valued attribute should be eliminated by moving it to another relation.
- ✓ Each compound attribute should be investigated for “if it **really** is compound” (of so, it should be split into several atomic attributes).

2NF: second normal form

Read and remember!

A relvar is in **2NF** if it satisfies 1NF and every its nonprime attribute is fully functionally dependent on the primary key.

In other words: there must not be an attribute, that is not a part of PK and still depends on a part of PK (not the entire PK).

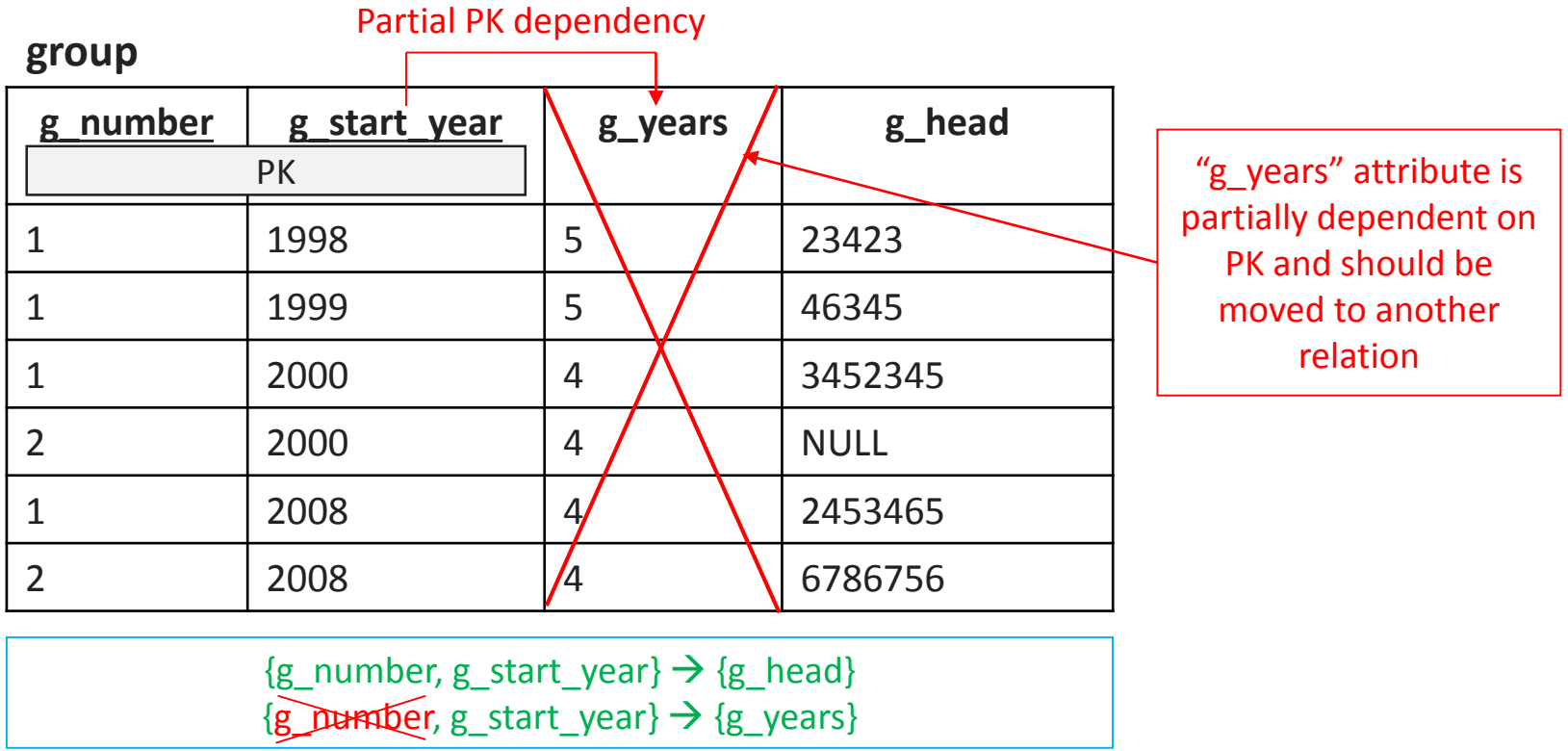
"Weak" definition

A relvar is in **2NF** if it satisfies 1NF and every its attribute is fully functionally dependent on any key.

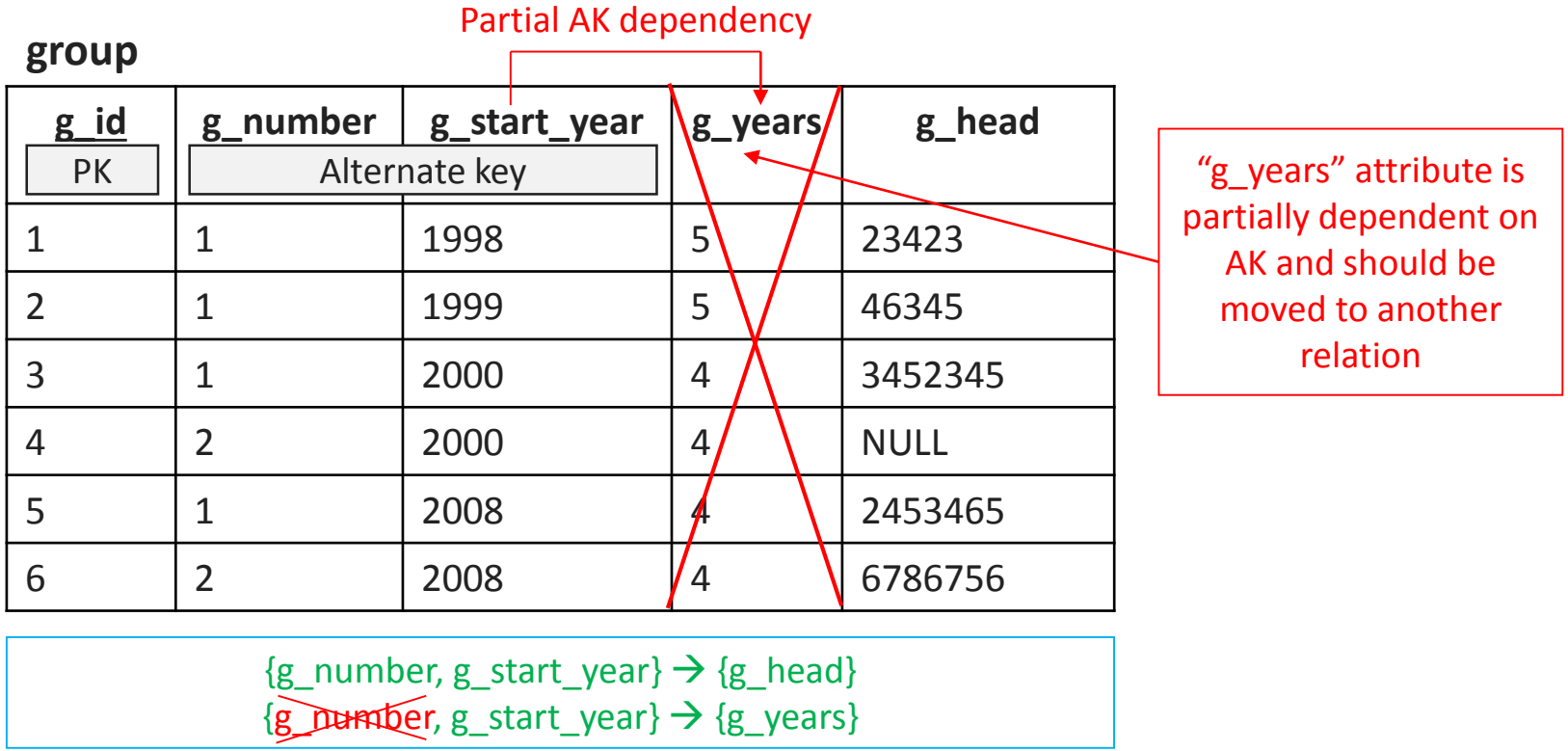
In other words: there must not be an attribute, that is not a part of any key and still depends on a part of any key (not the entire key).

"Strong" definition

2NF: second normal form (violation example, “weak” definition)



2NF: second normal form (violation example, “strong” definition)



2NF: second normal form (dealing with violations)

Before

group

Partial AK dependency

g_id (PK)	g_number (Alternate key)	g_start_year (Alternate key)	g_years	g_head
1	1	1998	5	23423
2	1	1999	5	46345
3	1	2000	4	3452345
4	2	2000	4	NULL

After

1-M relationship

education_length

el_start (PK)	el_years
1998	5
1999	5
2000	4

group

g_id (PK)	g_number (Alternate key)	g_start_year (Alternate key)	g_head
1	1	1998	23423
2	1	1999	46345
3	1	2000	3452345
4	2	2000	NULL

2NF: second normal form (conclusion)

- ✓ Always try using “strong” definition and find partial alternate keys dependencies.
- ✓ Each attribute that is partially dependent on alternate key should be moved to another relation. That part of alternate key the moved attribute was dependent on should become the PK of that new relation.

<epam>

Normal Forms – Part 1

Relational Databases Basics

TRAINING
CENTER

<epam>