‹epam›

Normal Forms – Part 3

Relational Databases Basics

TRAINING CENTER

‹epam›

Normal forms hierarchy

Sixth normal form (6NF)	
Domain-key normal form (DKNF)	Higher NFs
Fifth normal form (5NF)	
Fourth normal form (4NF)	
Boyce-Codd normal form (BCNF)	
Third normal form (3NF)	
Second normal form (2NF)	Lower NFs
First normal form (1NF)	
Zero normal form (0NF)	

Read and remember!
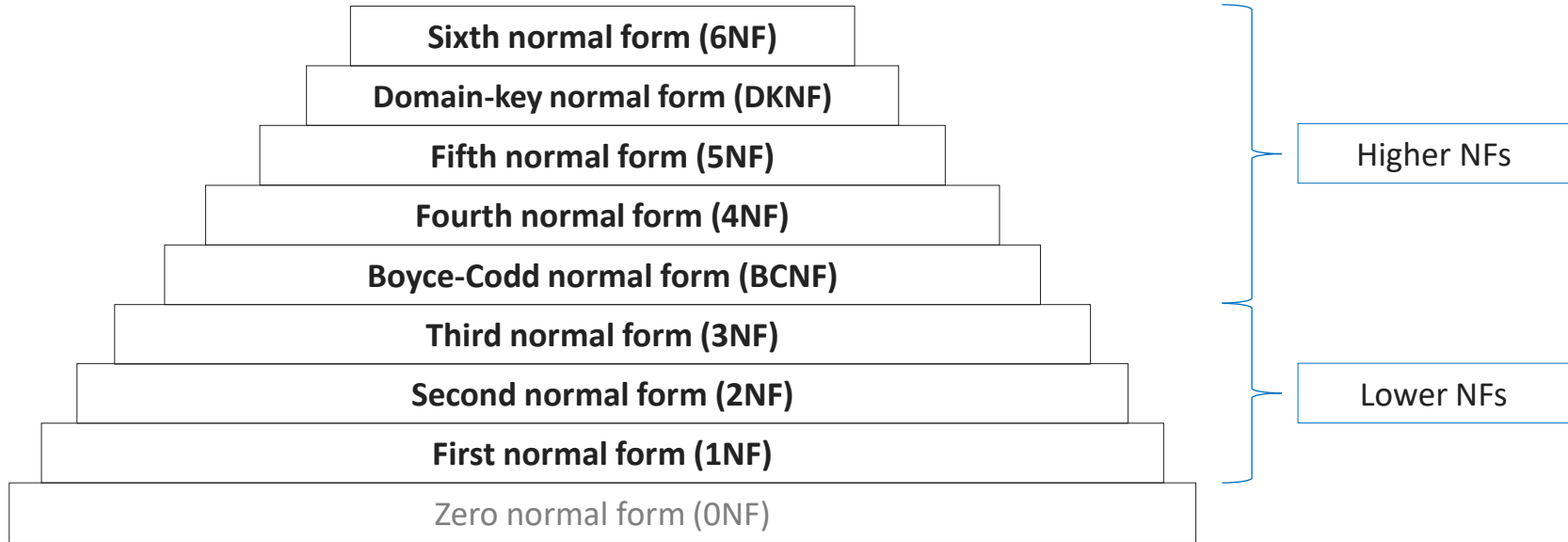
A relvar is in **5NF** if it satisfies 4NF, and with respect to a set F of functional, multivalued, and join dependencies if, for every nontrivial join dependency $JD(R_1, R_2, ..., R_n)$ in F^+ (that is, implied by F), every R_i is a superkey of the relvar.

*In other words: we have to know **each** candidate key and **each** join dependency of the relation, and we have to make sure that each projection (for each join dependency) contains a candidate key.*

5NF: fifth normal form (violation example)

workload

w_tutor	w_subject	w_faculty
(1) Ivanov I.I.	Mathematics	Exact sciences
(2) Ivanov I.I.	Informatics	Natural science
(3) Petrov P.P.	Mathematics	Natural science
(4) Sidorov S.S.	Informatics	Cybernetics
(5) Petrov P.P.	Physics	Exact sciences
(6) Petrov P.P.	Mathematics	Exact sciences
(7) Ivanov I.I.	Mathematics	Natural science

Join dependency:
JD({w_tutor, w_subject},
  {w_subject, w_faculty},
  {w_tutor, w_faculty})

**None of
(w_tutor, w_subject),
(w_subject, w_faculty),
(w_tutor, w_faculty) projections contains a potential key.**

There is a rule: "If a tutor deliver some subject S, and some faculty F has that subject S, and this tutor works on this faculty, he has to deliver the subject S on this faculty F".

a) There is "Mathematics" on "Exact sciences" faculty (1).
b) Petrov P.P. works on "Exact sciences" faculty (5).
c) Petrov P.P. delivers "Mathematics" (3).
FOLLOWS: Petrov P.P. has to deliver "Mathematics" on "Exact sciences" faculty (6).

5NF: fifth normal form (dealing with violations)

Before

workload

w_tutor	w_subject	w_faculty
PK		
Ivanov I.I.	Mathematics	Exact sciences
Ivanov I.I.	Informatics	Natural science
Petrov P.P.	Mathematics	Natural science
Sidorov S.S.	Informatics	Cybernetics
Petrov P.P.	Physics	Exact sciences
Petrov P.P.	Mathematics	Exact sciences
Ivanov I.I.	Mathematics	Natural science

After

No relationships here!

workload_ts

w_tutor	PK	w_subject
Ivanov I.I.		Mathematics
Ivanov I.I.		Informatics
Petrov P.P.		Mathematics
Sidorov S.S.		Informatics
Petrov P.P.		Physics

workload_sf

w_subject	PK	w_faculty
Mathematics		Exact sciences
Informatics		Natural science
Mathematics		Natural science
Informatics		Cybernetics
Physics		Exact sciences

workload_tf

w_tutor	PK	w_faculty
Ivanov I.I.		Exact sciences
Ivanov I.I.		Natural science
Petrov P.P.		Natural science
Sidorov S.S.		Cybernetics
Petrov P.P.		Exact sciences

5NF: fifth normal form (dealing with violations, more realistic)

After

tutor

t_id	PK	t_name
1		Ivanov I.I.
2		Petrov P.P.
3		Sidorov S.S.

subject

s_id	PK	s_name
1		Mathematics
2		Informatics
3		Physics

faculty

f_id	PK	f_name
1		Exact sciences
2		Natural science
3		Cybernetics

1-M

m2m_tutor_subject 1-M

t_id	PK	s_id
1		1
1		2
2		1
3		2
2		3

1-M   1-M

m2m_subject_faculty

s_id	PK	f_id
1		1
2		2
1		2
2		3
3		1

1-M   1-M

m2m_tutor_faculty

t_id	PK	f_id
1		1
1		2
2		2
3		3
2		1

5NF: fifth normal form (conclusion)

✔ 5NF (like 4NF) doesn't protect you from "local mistakes", but it keeps you from violating most "global rules".

✔ One of the most obvious evidence of 5NF violation is that a relation has a group of three or more attributes, and each and every attribute inside such group is dependent on each other.

✔ Most relations that satisfy 4NF also satisfy 5NF, because 5NF violation require specific subject matter rules.

DKNF: domain-key normal form

A relvar is in **DKNF** if all constraints and dependencies that should hold on the relation can be enforced simply by enforcing the domain constraints and key constraints on the relation.

In other words: any rules applicable to the relation are originated from its attributes properties, and there are no other "hidden" rules.

DKNF: domain-key normal form (violation example)

employee

PK		
e_name	**e_role**	
Ivanov I.I.	Coordinator	(P)
Ivanov I.I.	Programmer	(R)
Ivanov I.I.	Analyst	(R)
Petrov P.P.	Team-lead	(P)
Petrov P.P.	Programmer	(R)
Petrov P.P.	Architect	(R)
Ivanov I.I.	Manager	(P)

There is a rule: "Each employee may have any number of roles, but one position only".

DKNF: domain-key normal form (dealing with violations)

Before

After

employee

PK	
e_name	**e_role**
Ivanov I.I.	Coordinator
Ivanov I.I.	Programmer
Ivanov I.I.	Analyst
Petrov P.P.	Team-lead
Petrov P.P.	Programmer
Petrov P.P.	Architect
Ivanov I.I.	Manager

1-M relationship

employee_position

PK	
ep_name	**ep_position**
Ivanov I.I.	Coordinator
Petrov P.P.	Team-lead

employee_role

PK	
er_name	**er_role**
FK	
Ivanov I.I.	Programmer
Ivanov I.I.	Analyst
Petrov P.P.	Programmer
Petrov P.P.	Architect

DKNF: domain-key normal form (conclusion)

✓ Some relations may violate DKNF due to complicated subject matter rules.

✓ One of the most obvious evidence of DKNF violation is that the subject matter has such rules that enforce you to use complicated constraints beyond relational theory.

✓ Normalization to DKNF usually requires not a decomposition, but a complete rework of initial relation(s).

Read and remember!

A relvar is in **6NF** if it can't be nonloss decomposed at all, other than trivially — i.e., if and only if the only JDs to which it's subject are trivial ones.

In other words: while decomposing a relation into projections, at least one of such projections will be equivalent to the initial relation.

6NF: sixth normal form (violation example)

education_path

ep_student	ep_period	ep_faculty	ep_group
13452	01.01.2018-31.05.2018	Chemistry	1
13452	01.03.2018-31.05.2018	Chemistry	5
13452	01.06.2018-01.12.2018	Physics	5

The faculty and the group number may change independently, thus (obviously) we may store such data separately and bring this relation to a higher normal form.

6NF: sixth normal form (dealing with violations)

Before

education_path

ep_student	ep_period	ep_faculty	ep_group
13452	01.01.2018-31.05.2018	Chemistry	1
13452	01.03.2018-31.05.2018	Chemistry	5
13452	01.06.2018-01.12.2018	Physics	5

After

education_path_faculty

epf_student	epf_period	epf_faculty
13452	01.01.2018-31.05.2018	Chemistry
13452	01.06.2018-01.12.2018	Physics

education_path_group

epg_student	epg_period	epg_group
13452	01.01.2018-31.05.2018	1
13452	01.03.2018-01.12.2018	5

There is no possibility to decompose these relations any further (at least one of the new projections will be exactly the same as the initial one).

6NF: sixth normal form (conclusion)

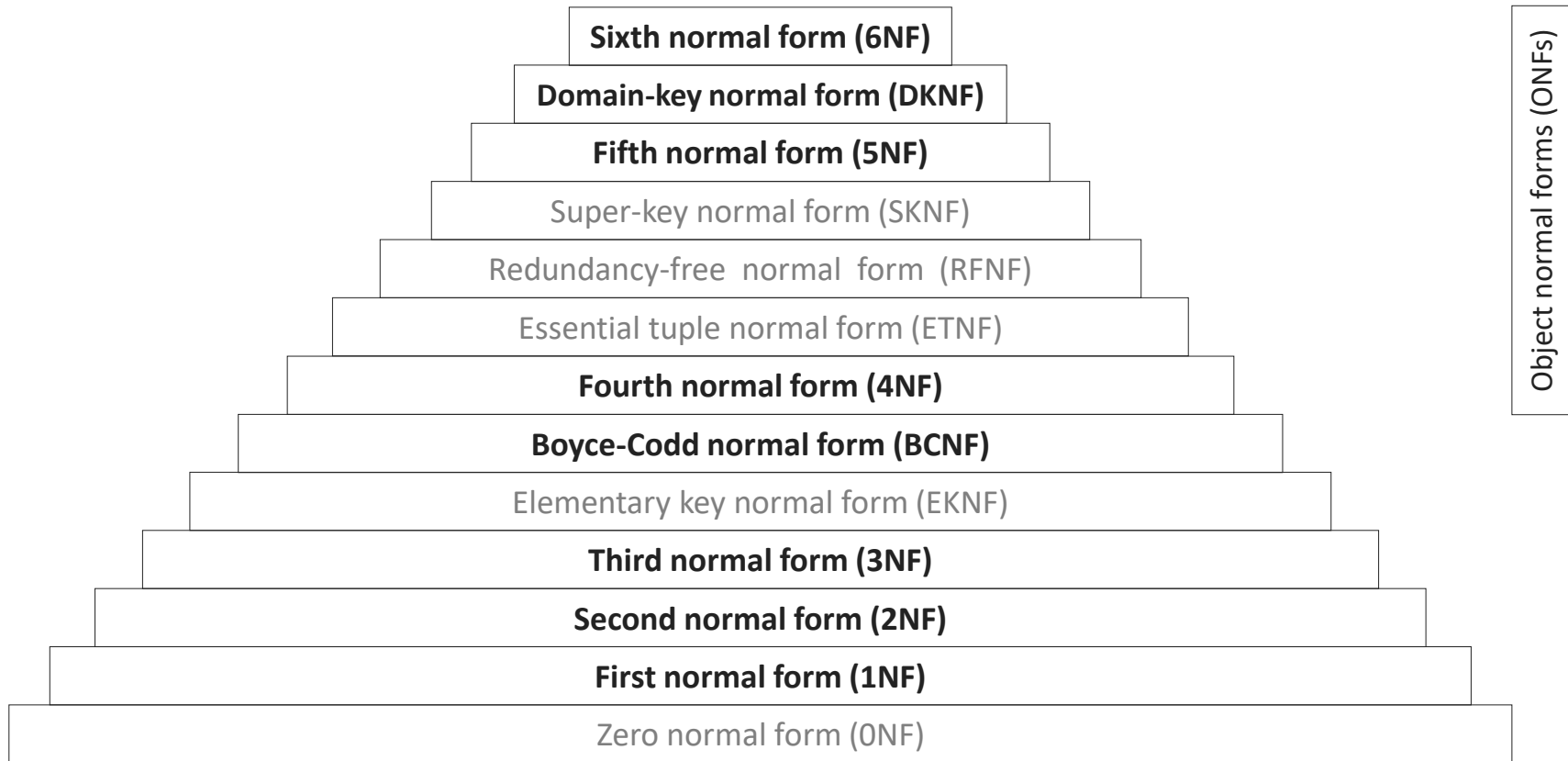✔ 6NF is actual for chronological relations (for any other relation 5NF is the ultimate one).

✔ One of the most obvious evidence of 6NF violation is that the relation has several independent attributes that are related to each other through the same chronological attribute.

✔ 6NF is the ultimate normal form. No further normalization is possible for any relation.

Normal forms hierarchy, one more quick overview…

Sixth normal form (6NF)

Domain-key normal form (DKNF)

Fifth normal form (5NF)

Super-key normal form (SKNF)

Redundancy-free normal form (RFNF)

Essential tuple normal form (ETNF)

Fourth normal form (4NF)

Boyce-Codd normal form (BCNF)

Elementary key normal form (EKNF)

Third normal form (3NF)

Second normal form (2NF)

First normal form (1NF)

Zero normal form (0NF)

Object normal forms (ONFs)

A small cheat-sheet

	0NF	1NF	2NF	3NF	AKNF	BCNF	4NF	ETNF	5NF	DKNF	6NF
There is a primary key	?	+	+	+	+	+	+	+	+	+	+
There are no multi-valued attributes	?	+	+	+	+	+	+	+	+	+	+
Each attribute is an atomic one	-	+	+	+	+	+	+	+	+	+	+
Each nonprime attribute is fully functionally dependent on the primary key	-	-	+	+	+	+	+	+	+	+	+
Each nonprime attribute is non-transitively dependent on the primary key	-	-	-	+	+	+	+	+	+	+	+
Any dependent attribute is a part of irreducible key	-	-	-	-	+	+	+	+	+	+	+
Each attribute is non-transitively dependent on the primary key	-	-	-	-	-	+	+	+	+	+	+
There are no non-trivial multi-valued dependencies	-	-	-	-	-	-	+	+	+	+	+
Every join dependency is based on a superkey	-	-	-	-	-	-	-	+	+	+	+
Every non-trivial join dependency is based on a candidate key	-	-	-	-	-	-	-	-	+	+	+
Each constraint is based on domains and keys constraints	-	-	-	-	-	-	-	-	-	+	+
Every join dependency is trivial	-	-	-	-	-	-	-	-	-	-	+

Normal Forms – Part 3

Relational Databases Basics