

System Analysis and Design

ໂດຍ: ອຈ ສິມມິດ ທຸມມາລີ

ບົດທີ 6

ແບບຈຳລອງຂໍ້ມູນ (Data Modeling)

ຂັ້ນຕອນການວິເຄາະຄວາມຕ້ອງການເພື່ອນຳໄປສູ່ ການອອກແບບຖານຂໍ້ມູນ

ແບບຈຳລອງຂໍ້ມູນເປັນເຕັກນິກທີ່ນຳມາໃຊ້ອະທິບາຍໂຄງສ້າງ
ແລະ ຄວາມສຳພັນລະຫວ່າງຂໍ້ມູນໃນລະບົບວ່າຂໍ້ມູນເລົ່ານັ້ນມີຄວາມສຳ
ພັນກັນແບບໃດ, ໂດຍໃນໄລຍະການວິເຄາະ, ນັກວິເຄາະລະບົບຈະຕ້ອງ
ສ້າງແບບຈຳລອງຂໍ້ມູນຕາມແນວຄິດ (Conceptual Data Models)
ເພື່ອສະແດງເຖິງພາບລວມຂອງລະບົບວ່າມີຂໍ້ມູນໃດແດ່ ແລະ ຕໍ່ມາກໍຈະ
ພັດທະນາລາຍລະອຽດຫຼາຍຂຶ້ນຈົນກາຍເປັນແບບຈຳລອງຂໍ້ມູນແບບຕັກ
ກະສາດ (Logical Data Models) ທີ່ສະແດງຂໍ້ມູນທັງລະບົບວ່າມີ
ຄວາມສຳພັນກັນແບບໃດ.

ການລວບລວມຂໍ້ມູນເພື່ອສ້າງແບບຈຳລອງຂໍ້ມູນຕາມ ແນວຄິດ

ແບບຈຳລອງຕາມແນວຄິດຈະຖືກສ້າງຂຶ້ນມາພ້ອມໆກັບ
ຂະບວນການວິເຄາະຫາຄວາມຕ້ອງການທີ່ໄດ້ມາຈາກການເຂົ້າໄປ
ສືບຄົ້ນຫາຂໍ້ແທ້ຈິງເພື່ອໃຫ້ໄດ້ມາເຊິ່ງຄວາມຕ້ອງການຂອງຜູ້ໃຊ້
ຈາກນັ້ນກໍຈະນຳມາວິເຄາະເພື່ອສ້າງເປັນຂໍ້ກຳນົດຄວາມຕ້ອງການ
ຂອງລະບົບໃໝ່ດ້ວຍການສ້າງແບບຈຳລອງຂະບວນການຂຶ້ນມານັ້ນ
ກໍຄື ແຜນວາດການໄຫຼຂໍ້ມູນທີ່ສະແດງເຖິງຄວາມສຳພັນລະຫວ່າງ
Process ກັບ ຂໍ້ມູນໂດຍຂໍ້ມູນທີ່ໄດ້ຈັດເກັບລົງໃນ Data
Store ເລົ່ານີ້ຈະຖືກບັນທຶກລົງໃນຖານຂໍ້ມູນຢ່າງເປັນລະບົບ.

ERD

ການວິເຄາະເພື່ອໃຫ້ໄດ້ ERD (Entity Relationship Diagram) ຈະມີພື້ນຖານຫຼັກຢູ່ 3 ປະການຄື: Entity, Relationship ແລະ Attributes

Entity

Entity ໝາຍເຖິງສິ່ງຕ່າງໆໃນຖານຂໍ້ມູນທີ່ອ້າງອີງເຖິງ, ສິ່ງເລົ່ານີ້ອາດຈະເປັນໄດ້ທັງຮູບປະທຳເຊັ່ນ: ຄົນ, ສັດ, ສິ່ງຂອງ ຫຼື ເປັນໄປໄດ້ທັງນາມມະທຳເຊັ່ນ: ຄວາມສຳນານ, ອາຊີບ ເປັນຕົ້ນ ດັ່ງນັ້ນ Entity ກໍຄືກຸ່ມຂໍ້ມູນປະເພດດຽວກັນ, ເຊັ່ນວ່າ ຖ້າເວົ້າເຖິງ Entity ພະນັກງານຈະໝາຍເຖິງກຸ່ມຄົນທັງໝົດທີ່ເປັນພະນັກງານ.

ສະນັ້ນ Entity ເປັນໄດ້ທັງບຸກຄົນ (Persons), ສະຖານທີ່ (Place), ວັດຖຸ (Objects), ເຫດການ (Events) ຫຼື ແນວຄິດ (Concepts) ເຊິ່ງມີສັນຍາລັກດັ່ງລຸ່ມນີ້:

CUSTOMER

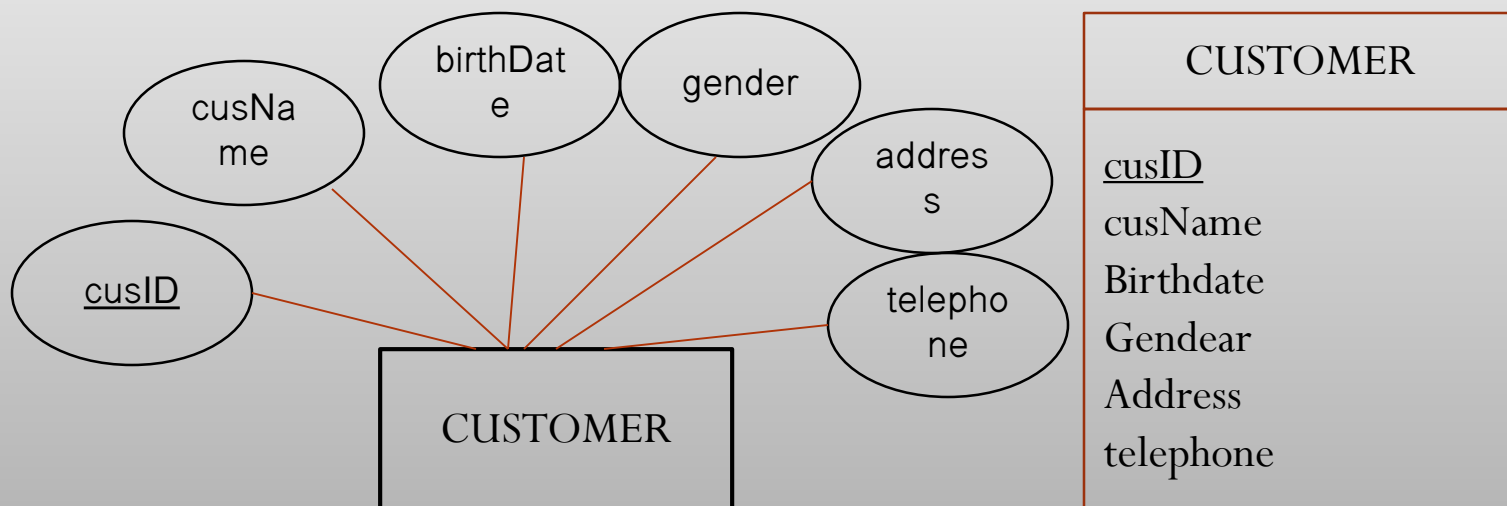
CONTRACT

ຕົວຢ່າງ: Entity (Table)

Name	AGE	ADDRESS	JOB
ສຸດາພອນ ຈັນມາ	23	123 ເມືອງໄຊທານີ	ນັກຂຽນໂປຣແກຣມ
ສຸພາບ ມາດີ	45	123 ເມືອງ ສີໂຄດ	ນັກວິເຄາະລະບົບ
ຈັນທິດາ ມາກ່ອນ	30	56/4 ເມືອງ ໄຊເສດຖາ	ນັກຂຽນໂປຣແກຣມ

Attribute (Field)

ໝາຍເຖິງສິ່ງທີ່ໃຊ້ບອກອົງປະກອບຂອງ Entity ເຊິ່ງແຕ່ລະ Entity ຈະມີອົງປະກອບສະເພາະຕົວເຊັ່ນວ່າ Entity ພະນັກງານຈະປະກອບດ້ວຍ Attribute ຕ່າງໆຄື: ລະຫັດປະຈຳຕົວ, ຊື່, ນາມສະກຸນ, ເພດ, ອາຍຸ, ພະແນກທີ່ສັງກັດ ເປັນຕົ້ນ. ເຊິ່ງມີສັນຍາລັກດັ່ງນີ້



ຕົວຢ່າງ: Attribute (Field)

Attribute

Name	AGE	ADDRESS	JOB
ສຸດາພອນ ຈັນມາ	23	123 ເມືອງໄຊທານີ	ນັກຂຽນໂປຣແກຣມ
ສຸພາບ ມາດີ	45	123 ເມືອງ ສີໂຄດ	ນັກວິເຄາະລະບົບ
ຈັນທິດາ ມາກ່ອນ	30	56/4 ເມືອງ ໄຊເສດຖາ	ນັກຂຽນໂປຣແກຣມ

Attribute value

ຊະນິດຂອງຂໍ້ມູນ (Data Type)

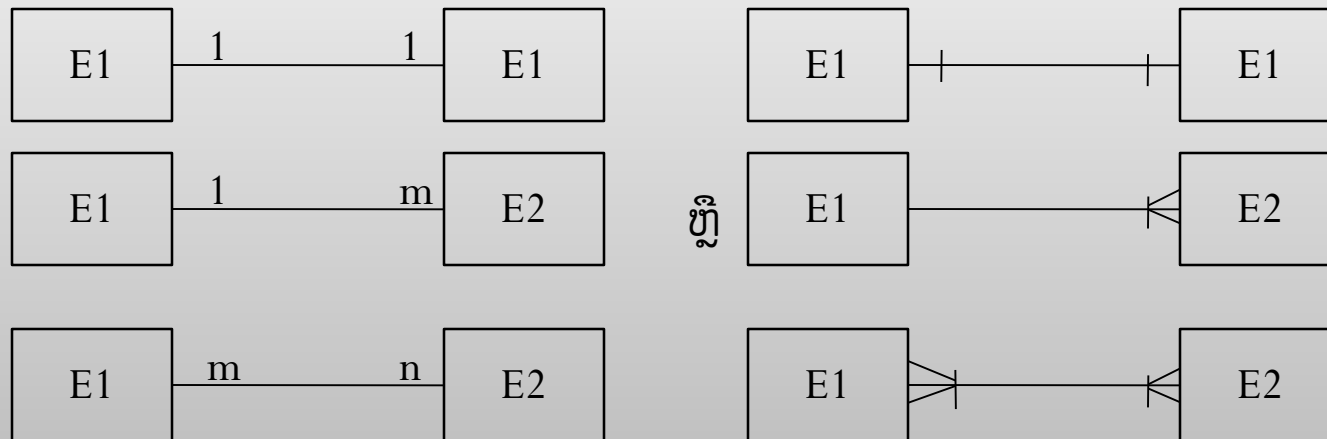
ຊະນິດຂໍ້ມູນ	ຄວາມໝາຍ
NUMBER	ຂໍ້ມູນເປັນຕົວເລກ (ເລກຈຳນວນຈິງ ແລະ ເລກຈຳນວນຖ້ວນ)
TEXT	ຂໍ້ມູນທີ່ເປັນຕົວອັກສອນ ແລະ ຕົວເລກ, ແຕ່ຖ້າບັນທຶກເປັນຕົວເລກຈະບໍ່ສາມາດນຳມາຄິດໄລ່ໄດ້
MEMO	ຂໍ້ມູນທີ່ຄ້າຍຄືກັບ Text ແຕ່ຈະເໝາະສົມກັບການບັນທຶກຂໍ້ຄວາມຕ່າງໆທີ່ເຮົາບໍ່ສາມາດກຳນົດຂະໜາດຂອງຂໍ້ມູນ
DATE	ຂໍ້ມູນວັນທີ່ເຊິ່ງເປັນໄປຕາມຮູບແບບວັນທີ່ກຳນົດ
TIME	ຂໍ້ມູນເວລາ ເຊິ່ງເປັນໄປຕາມຮູບແບບເວລາທີ່ກຳນົດ
YES/NO	ຂໍ້ມູນທີ່ເປັນແບບ Logic ເຊິ່ງມີຄ່າເປັນຈິງ ຫຼື ບໍ່ຈິງ
IMAGE/PICTURE	ຂໍ້ມູນທີ່ເປັນຮູບພາບ

Relationship

Relationship ແມ່ນສິ່ງທີ່ໃຊ້ສະແດງເຖິງຄວາມສໍາພັນລະຫວ່າງ Entity 2 Entity ຫຼື ອາດຫຼາຍກ່ວາ 2 Entity ກໍ່ໄດ້, ແຕ່ໃນການອອກແບບຖານຂໍ້ມູນເພິ່ນພິຈາລະນາຄວາມສໍາພັນທີ່ເກີດຂຶ້ນລະຫວ່າງ 2 Entity. ສໍາລັບ Entity ແຕ່ລະຕົວອາດເກີດຄວາມສໍາພັນໄດ້ຫຼາຍກ່ວາ 1 ຄວາມສໍາພັນ.

ປະເພດຄວາມສໍາພັນ (Relationship)

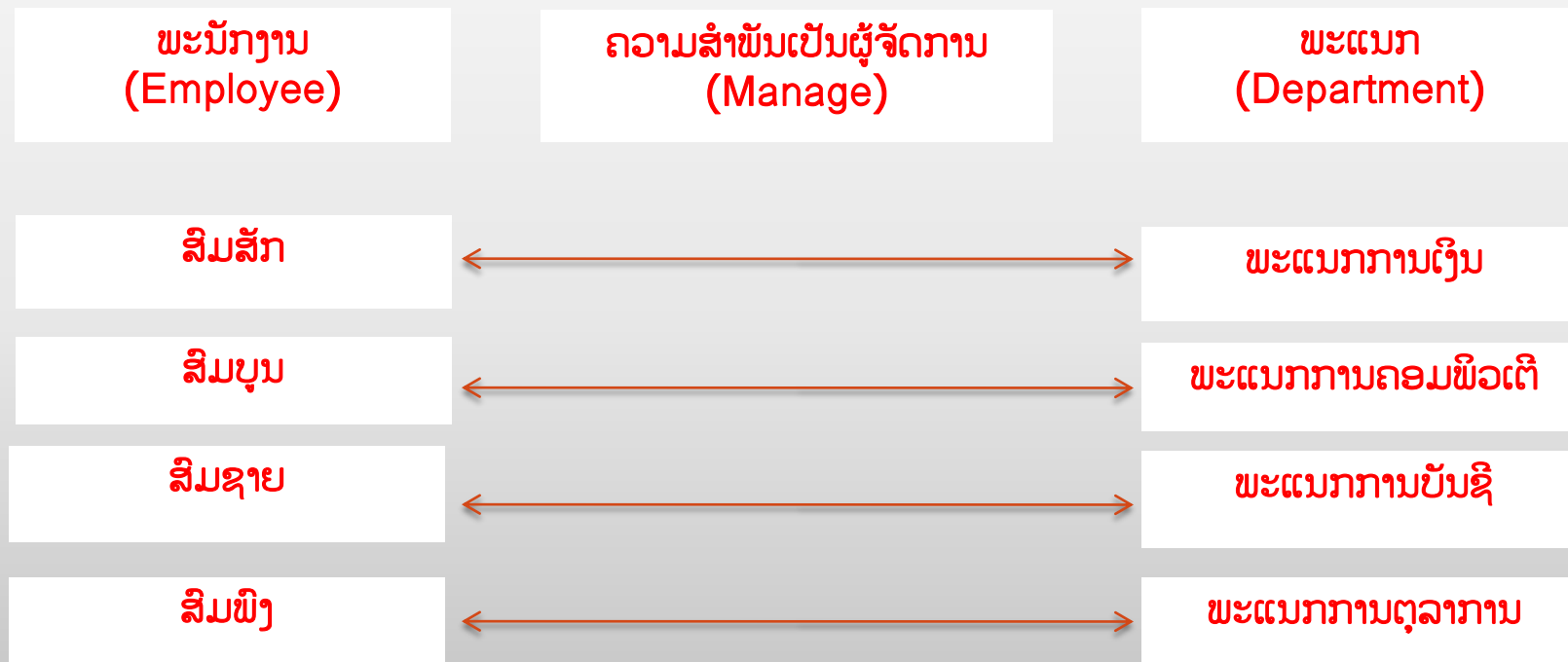
- ຄວາມສໍາພັນແບບໜຶ່ງຕໍ່ໜຶ່ງ (One-to-One Relationship)
- ຄວາມສໍາພັນແບບໜຶ່ງຕໍ່ຫຼາຍ (One-to-Many Relationship)
- ຄວາມສໍາພັນແບບຫຼາຍຕໍ່ຫຼາຍ (Many-to-Many Relationship)



ຄວາມສຳພັນແບບ 1 : 1

ຖ້າ Entity E1 ມີຄວາມສຳພັນກັບ Entity E2 ແບບ ໜຶ່ງຕໍ່ໜຶ່ງ, ນັ້ນໝາຍຄວາມວ່າສະມາຊິກຂອງ Entity E1 ໜຶ່ງລາຍການຈະມີຄວາມສຳພັນກັບສະມາຊິກໃນ Entity E2 ໜຶ່ງລາຍການເທົ່ານັ້ນ.

ຕົວຢ່າງ: ຄວາມສໍາພັນເປັນຜູ້ຈັດການ



ຕົວຢ່າງ: ຄວາມສໍາພັນລະຫວ່າງ 2 Entity ແບບ 1:1

ພະນັກງານ

ລະຫັດພ/ງ	ຊື່	ນາມສະກຸນ	ທີ່ຢູ່	ໂທລະສັບ
1001	Micheal	Suyama	Coventry House, London	(71)555-7773
1002	Laura	Callahan	4726-11 th Ave.N.E. Seattle	(71)555-4848
1003	Robert	King	Edgeham Hollow, London	(206)555-1189
1004	Anne	Dodsworth	7 Hounstood Rd., London	(71)555-4444

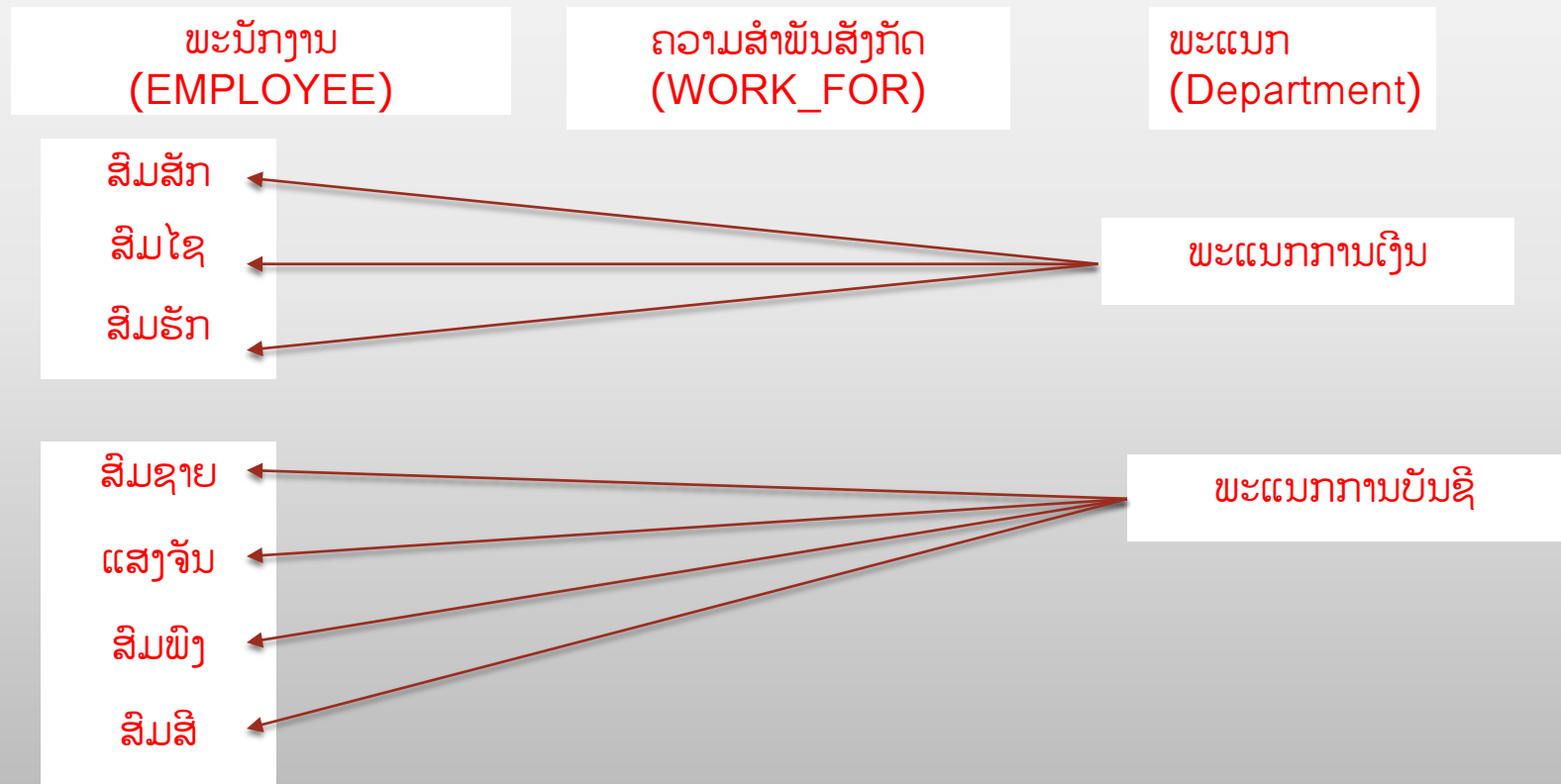
ຂໍ້ມູນສ່ວນຕົວ

ລະຫັດພ/ງ	ເງິນເດືອນ	ເລກບັດປະກັນສັງຄົມ
1001	25000	324-45-6400
1002	20000	245-37-5600
1003	34000	459-65-8522
1004	30000	895-56-1200

ຄວາມສໍາພັນແບບ 1 : N

ຖ້າ Entity E1 ມີຄວາມສໍາພັນກັບ Entity E2 ແບບ ໜຶ່ງຕໍ່ຫຼາຍ, ນັ້ນໝາຍຄວາມວ່າສະມາຊິກຂອງ Entity E1 ໜຶ່ງລາຍການຈະມີຄວາມສໍາພັນກັບສະມາຊິກໃນ Entity E2 ໄດ້ຫຼາຍກວ່າໜຶ່ງລາຍການ, ແຕ່ສະມາຊິກໃນ Entity E2 ໜຶ່ງລາຍການຈະສາມາດມີຄວາມສໍາພັນກັບສະມາຊິກໃນ Entity E1 ໄດ້ພຽງແຕ່ 1 ລາຍການເທົ່ານັ້ນ.

ຕົວຢ່າງ ຄວາມສໍາພັນສັງກັດ



ຕົວຢ່າງຄວາມສຳພັນລະຫວ່າງ 2 Entity ແບບ 1:N

ລູກຄ້າ

ລະຫັດລູກຄ້າ	ຊື່ບໍລິສັດ	ທີ່ຢູ່ບໍລິສັດ
1001	Let's Stop N Shop	87 Polk St, San Francisco
1002	B's Beverages	Fauntleroy Cirus, London
1003	Chop-suey chinese	Guaptstr. 29, Bern

1

ການສັ່ງຊື້

ລະຫັດສັ່ງຊື້	ລະຫັດລູກຄ້າ	ລະຫັດສິນຄ້າ
12010	1001	BE-203
12015	1003	ME-010
12016	1002	SE-057
12025	1003	SE-057

N

ຄວາມສໍາພັນແບບຫຼາຍຕໍ່ຫຼາຍ (N:M)

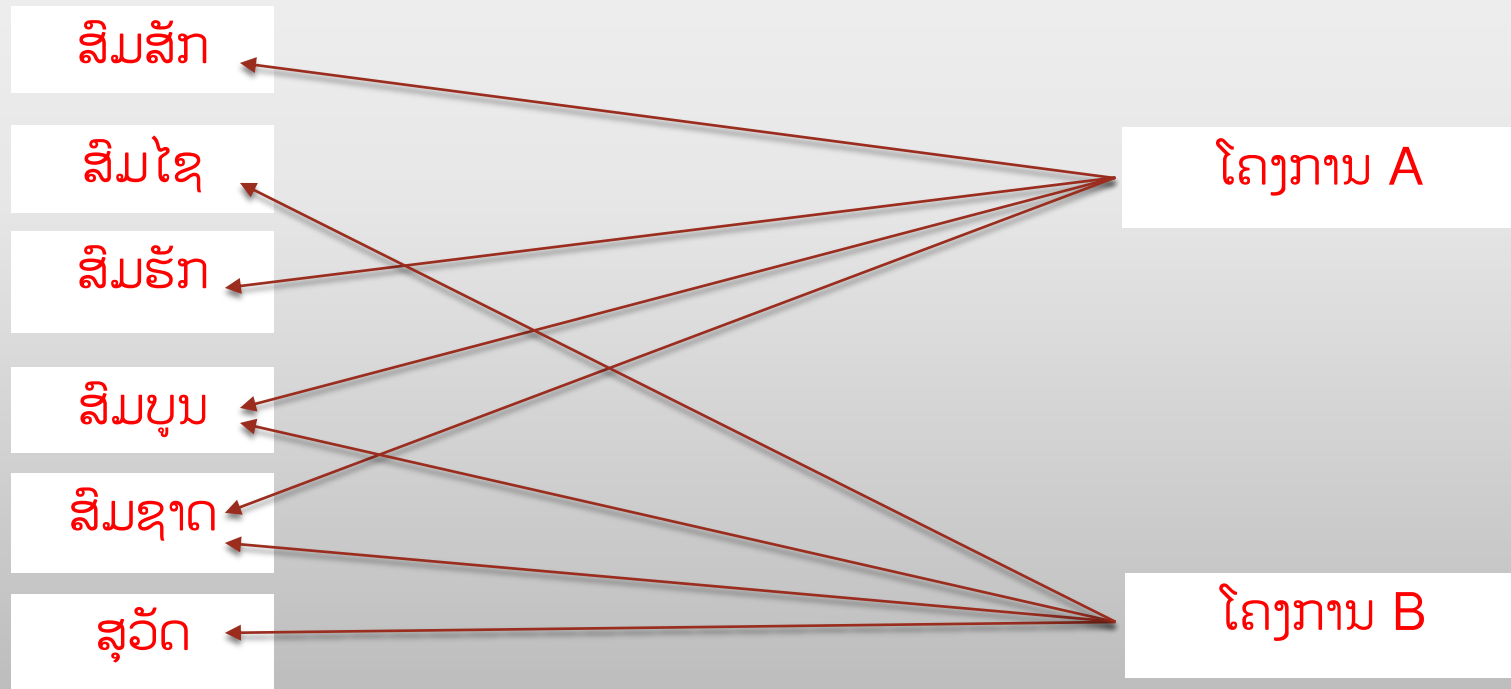
ຖ້າ Entity E1 ມີຄວາມສໍາພັນແບບຫຼາຍຕໍ່ຫຼາຍກັບ Entity E2 , ນັ້ນໝາຍຄວາມວ່າສະມາຊິກໃນ Entity E1 ໜຶ່ງລາຍການຈະມີຄວາມສໍາພັນກັບສະມາຊິກໃນ Entity E2 ໄດ້ຫຼາຍກວ່າໜຶ່ງລາຍການ ແລະ ສະມາຊິກໃນ Entity E2 ໜຶ່ງລາຍການຈະມີຄວາມສໍາພັນກັບສະມາຊິກໃນ Entity E1 ໄດ້ຫຼາຍກວ່າໜຶ່ງລາຍການເຊັ່ນກັນ.

ຕົວຢ່າງ ຄວາສຳພັນການເຮັດວຽກ

ພະນັກງານ
(EMPLOYEE)

ຄວາມສຳພັນການເຮັດວຽກ
(WORK_ON)

ໂຄງການ
(PROJECT)



ຕົວຢ່າງ ຄວາມສໍາພັນລະຫວ່າງ 2 Entity ແບບ N:M

ລູກຄ້າ

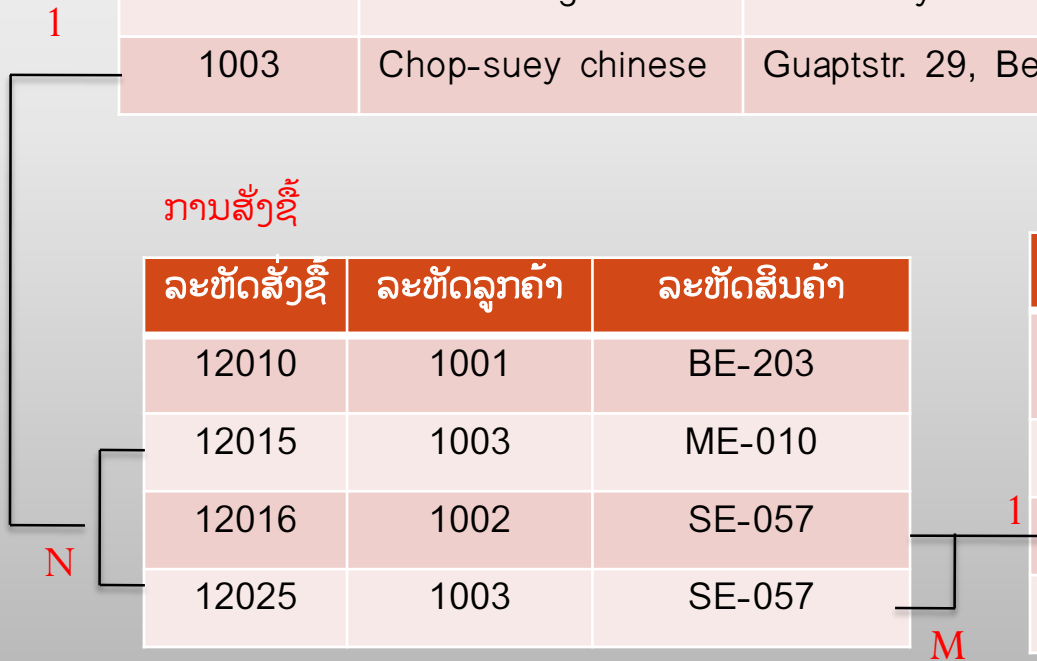
ລະຫັດລູກຄ້າ	ຊື່ບໍລິສັດ	ທີ່ຢູ່ບໍລິສັດ
1001	Let's Stop N Shop	87 Polk St, San Francisco
1002	B's Beverages	Fauntleroy Cirus, London
1003	Chop-suey chinese	Guaptstr. 29, Bern

ການສັ່ງຊື້

ລະຫັດສັ່ງຊື້	ລະຫັດລູກຄ້າ	ລະຫັດສິນຄ້າ
12010	1001	BE-203
12015	1003	ME-010
12016	1002	SE-057
12025	1003	SE-057

ສິນຄ້າ

ລະຫັດສິນຄ້າ	ຊື່ສິນຄ້າ	ຈຳນວນ
ME-010	Boston Crab Meat	120
BE-115	Ipoh Coffee	15
SE-057	Konbu	24
BE-203	Outback Lager	30



ຕົວຢ່າງ

ຈາກການວິເຄາະແບບຟອມການເຊົ່າລົດຂອງລູກຄ້າສາມາດນຳມາສ້າງເປັນ ER Diagram ໄດ້ໂດຍພົບວ່າ Entity ຕ່າງໆທີ່ຖືກກຳນົດຢູ່ໃນແຜນວາດໄດ້ກໍ່ໃຫ້ເກີດແນວຄິດຂອງກຸ່ມຂໍ້ມູນຕ່າງໆທີ່ກ່ຽວຂ້ອງກັບຂະບວນການນັ້ນໆເຊິ່ງປະກອບດ້ວຍ:

Customer ໝາຍເຖິງ ລູກຄ້າ

Contract ໝາຍເຖິງ ສັນຍາເຊົ່າ

Car ໝາຍເຖິງ ລົດ

Carrent_Item ໝາຍເຖິງ ລາຍການເຊົ່າລົດ

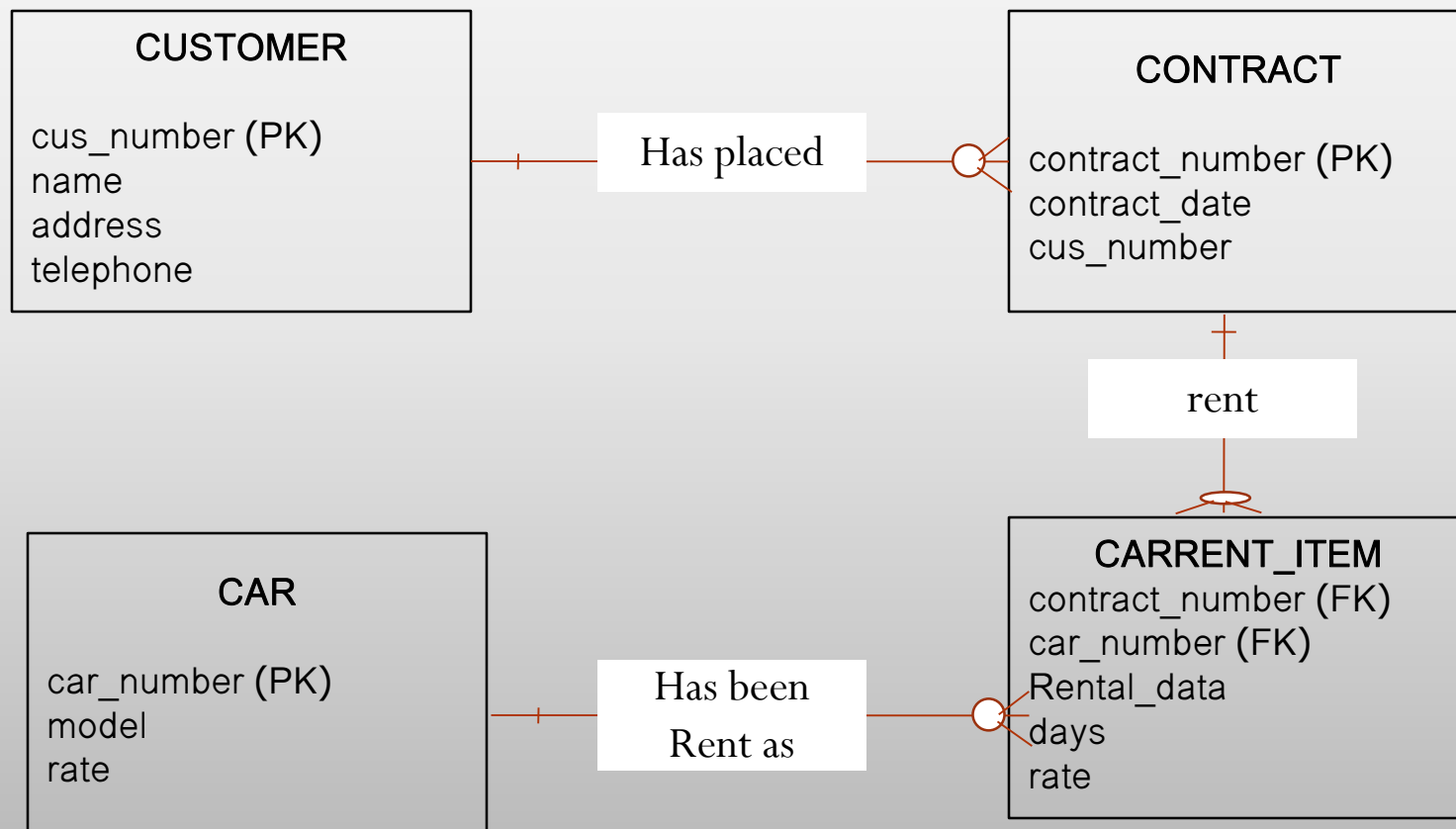
ເຊິ່ງສາມາດອະທິບາຍເຖິງຄວາມສຳພັນລະຫວ່າງ Entity ໄດ້ດັ່ງນີ້:

ລູກຄ້າໜຶ່ງຄົນ ເຮັດສັນຍາເຊົ່າລົດໄດ້ຫຼາຍສະບັບ

ສັນຍາເຊົ່າລົດໜຶ່ງສະບັບ ສາມາດເຊົ່າລົດໄດ້ຫຼາຍຄັນ

ລົດໜຶ່ງຄັນ ຖືກເຊົ່າໄດ້ຫຼາຍຄັ້ງ

ER ການເຊົ່າລົດຂອງລູກຄ້າ



ຄວາມສົມດູນລະຫວ່າງ ER ແລະ DFD

ເຖິງແມ່ນວ່າແຜນວາດການໄຫຼຂໍ້ມູນເປັນແບບຈຳລອງທີ່ມັ່ງເນັ້ນເຖິງຂະບວນການ ແລະ ຂໍ້ມູນ ໂດຍມີຈຸດປະສົງເພື່ອໃຫ້ຮູ້ວ່າມີຂໍ້ມູນໃດແດ່ທີ່ນຳມາໃຊ້ ແລະ ມີຂໍ້ມູນໃດແດ່ທີ່ຖືກສ້າງຂຶ້ນໂດຍ Process ນັ້ນໆ ແລະ ສະຖານທີ່ໃດເປັນທີ່ເກັບຂໍ້ມູນເລົ່ານັ້ນ ເຊິ່ງສ່ວນປະກອບຂໍ້ມູນໃນແຜນວາດການໄຫຼຂໍ້ມູນເລົ່ານີ້ຈະຕ້ອງສອດຄ່ອງກັບການຈັດເກັບຂໍ້ມູນໃນແຜນວາດ ER. ໝາຍວ່າ Data Store ທີ່ຢູ່ໃນ DFD ຈະຕ້ອງປະກົດຢູ່ໃນ ER, ບໍ່ດັ່ງນັ້ນກໍຈະເກີດຄວາມບໍ່ສົມດູນໃນລະບົບ. ເຊິ່ງຂໍ້ຜິດພາດດັ່ງກ່າວເຮັດໃຫ້ສາມາດເວົ້າໄດ້ວ່າມີການຈັດເກັບຂໍ້ມູນທີ່ບໍ່ຈຳເປັນ. ສຳລັບວິທີການກວດສອບຄວາມສົມດູນລະຫວ່າງ ER ແລະ DFD ຕ້ອງພິຈາລະນາຈຳນວນ Data Store ໃນ DFD Level 1 (Diagram 0) ຈະຕ້ອງມີຈຳນວນເທົ່າກັບ Entity ໃນ ER.

Key

ເພື່ອເພີ່ມປະສິດທິພາບໃນການຈັດເກັບຂໍ້ມູນຄວນກຳນົດ Key ໃຫ້ກັບ Entity ທີ່ໃຊ້ຈຳແນກ Record ແລະ ກຳນົດຄວາມສຳພັນລະຫວ່າງ Entity. Key ທີ່ໃຊ້ກັນທົ່ວໄປ ປະກອບມີ:

- Primary Key (Key ຫຼັກ)
- Foreign Key (key ນອກ)

Primary Key

Attribute ໃດທີ່ມີຄຸນສົມບັດເປັນ Primary key ຄ່າໃນ Attribute ນັ້ນຕ້ອງບໍ່ສາມາດຊ້ຳກັນ (unique) ໄດ້ ແລະ ບໍ່ສາມາດເປັນຫວ່າງເປົ່າໄດ້.

ຕົວຢ່າງ: Entity: ພະນັກງານ

ລະຫັດພະນັກງານ	ຊື່	ນາມສະກຸນ	ທີ່ຢູ່	ໂທລະສັບ
1001	ສິມມິດ	ທຸມມາລີ	ມຊ (ດົງໂດກ)	020 5720450
1002	ສຸລິດ	ແສງມະໂນທຳ	ມຊ (ດົງໂດກ)	020 6255818
1003	ອາມອນ	ຈັນທະພາວົງ	ມຊ (ດົງໂດກ)	020 7600972
1004	ພູທອນ	ວົງປະສິດ	ມຊ (ດົງໂດກ)	020 7525023

ເປັນ Primary Key

Foreign Key

ເປັນ Key ທີ່ໃຊ້ເຊື່ອມໂຍງລະຫວ່າງ Entity ທີ່ມີຂໍ້ມູນກ່ຽວຂ້ອງກັນ

ຕົວຢ່າງ Entity ລູກຄ້າຈະມີ Attribute ລະຫັດລູກຄ້າເປັນ key ຫຼັກ, ເຮົາສາມາດໃຫ້ລະຫັດລູກຄ້າໃນ Entity ລູກຄ້າເຊື່ອມໂຍງກັບລະຫັດລູກຄ້າໃນ Entity ການສັ່ງຊື້ເພື່ອຈະໄດ້ຮູ້ຊື່ ແລະ ທີ່ຢູ່ຂອງລູກຄ້າທີ່ສັ່ງຊື້ສິນຄ້ານັ້ນ, ໃນກໍລະນີນີ້ Attribute ລະຫັດລູກຄ້າໃນ Entity ການສັ່ງຊື້ຈະມີຄຸນສົມບັດເປັນ Foreign key, ສ່ວນລະຫັດລູກຄ້າໃນ Entity ລູກຄ້າຈະມີຄຸນສົມບັດເປັນ Primary key

ຕົວຢ່າງ: Entity: ລູກຄ້າ ແລະ Entity: ການສັ່ງຊື້

ລູກຄ້າ

ລະຫັດລູກຄ້າ	ຊື່ບໍລິສັດ	ທີ່ຢູ່
1001	Inter com	ກຳແພງນະຄອນ
1002	Huawei	ກຳແພງນະຄອນ
1003	SP computer	ກຳແພງນະຄອນ

1

ການສັ່ງຊື້

ລະຫັດສັ່ງຊື້	ລະຫັດລູກຄ້າ	ລະຫັດສິນຄ້າ
12010	1001	BE-203
12015	1003	ME-010
12016	1002	BE-057
12025	1003	BE-057

N

ພົດຈະນານຸກົມຂໍ້ມູນ (Data Dictionary)

ພົດຈະນານຸກົມຂໍ້ມູນແມ່ນເອກະສານທີ່ໃຊ້ອະທິບາຍລາຍ

ລະອຽດໂຄງສ້າງແຟ້ມຂໍ້ມູນລວມໄປເຖິງລາຍການຂອງຂໍ້ມູນຕ່າງໆ

ເຊັ່ນ: Relation Name, Attribute, Description, Type,

PK, FK ແລະ Reference

ຕົວຢ່າງວັດຈະນານຸກົມຂໍ້ມູນ

Entity: Contract

Attribute	Description	Type	PK	FK	Reference
Contract_Number	ລະຫັດສັນຍາເຊົ່າ	char(8)	Yes		
Customer_number	ລະຫັດລູກຄ້າ	char(8)		Yes	Customer
Contract_Date	ວັນທີເດືອນປີເຮັດສັນຍາ	Date			

Entity: Carrent-Item

Attribute	Description	Type	PK	FK	Reference
Contract_Number	ລະຫັດສັນຍາເຊົ່າ	char(8)		Yes	Contract
Car_number	ລະຫັດລົດ	char(8)		Yes	Car
Rent_Date	ວັນທີເດືອນປີເຊົ່າ	Date			
Days	ຈຳນວນວັນ	Number			

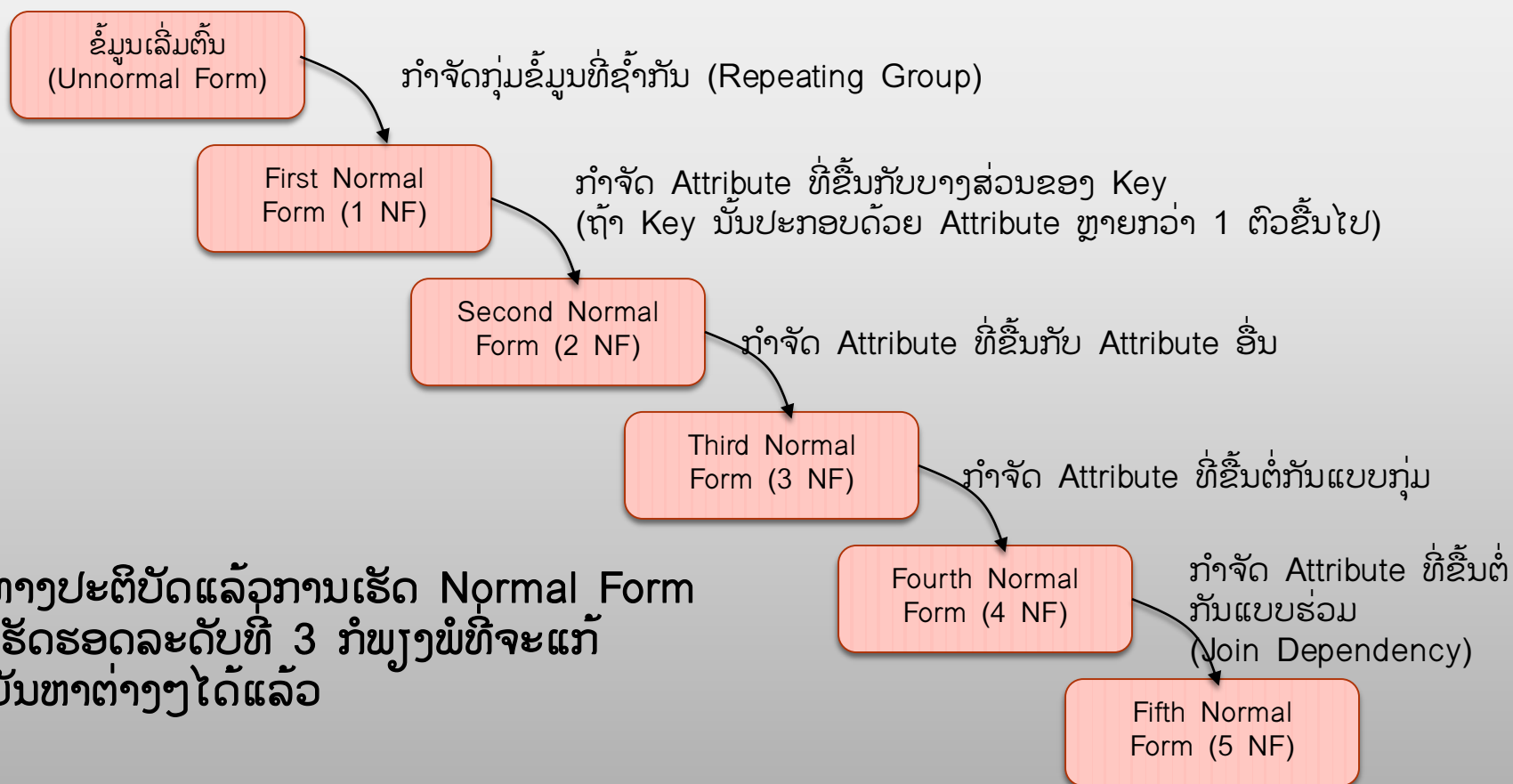
Normalization

ຂໍ້ມູນທີ່ເກັບລວບລວມມານັ້ນອາດຈະມີຄວາມຊໍ້າຊ້ອນຫຼາຍເກີນ
ໄປບໍ່ເໝາະສົມທີ່ຈະນໍາເອົາມາໃຊ້ໂດຍກົງຈຶ່ງໄດ້ເກີດທິດສະດີການເຮັດ
Normalization ເພື່ອປັບໂຄງສ້າງຂອງຂໍ້ມູນໃຫ້ຢູ່ໃນຮູບປົກກະຕິ
(Normal Form) ທີ່ງ່າຍຕໍ່ການນໍາໄປໃຊ້ວຽກ ແລະ ມີບັນຫາໜ້ອຍທີ່
ສຸດບໍ່ວ່າຈະເປັນບັນຫາການຊໍ້າຊ້ອນຂອງຂໍ້ມູນ ຫຼື ບັນຫາຄວາມຜິດ
ປົກກະຕິທີ່ອາດຈະເກີດຂຶ້ນຈາກການເພີ່ມ, ການລຶບ ຫຼື ການປ່ຽນແປງຂໍ້
ມູນ.

ຜົນປະໂຫຍດທີ່ໄດ້ຈາກການເຮັດ Normalization

- ຫຼຸດຜ່ອນບັນຫາການເກັບຂໍ້ມູນຊ້ຳກັນໃນຫຼາຍ Table ເຊິ່ງຈະເຮັດໃຫ້ເບື້ອງເນື້ອທີ່ຈັດເກັບໂດຍບໍ່ຈຳເປັນ
- ຫຼຸດຜ່ອນຄວາມຂັດແຍ່ງຂອງຂໍ້ມູນເນື່ອງຈາກການແກ້ໄຂຂໍ້ມູນແຕ່ລະຄັ້ງຈະເຮັດພຽງບ່ອນດຽວໃນຖານຂໍ້ມູນ
- ເຮັດໃຫ້ແກ້ໄຂໂຄງສ້າງຂອງ Table ໄດ້ງ່າຍຂຶ້ນ ແລະ ມີຜົນກະທົບກັບ Table ອື່ນໆທີ່ກ່ຽວຂ້ອງໜ້ອຍທີ່ສຸດ

ຂັ້ນຕອນການເຮັດ Normalization



ໃນທາງປະຕິບັດແລ້ວການເຮັດ Normal Form
ຈະເຮັດຮອດລະດັບທີ 3 ກໍພຽງພໍທີ່ຈະແກ້
ໄຂບັນຫາຕ່າງໆໄດ້ແລ້ວ

1NF (First Normal Form)

Entity ໃດຈະຖືວ່າຢູ່ໃນ 1NF ຫຼື ບໍ່ ຈະຕ້ອງພິຈາລະນາເບິ່ງ
ທຸກໆ Field ໃນ Entity ນັ້ນ ຕ້ອງບໍ່ມີ Attribute (Field)
ໃດ Attribute ນຶ່ງທີ່ມີລັກສະນະເປັນ Multivalued (ໝາຍວ່າ
Attribute ດຽວແຕ່ມີຫຼາຍຄ່າ)

ຕົວຢ່າງ (Entity ບໍ່ຜ່ານ 1NF)

Entity: Car

SerialNo	Brand	Model	Color
1ຂ-2778	Benz	E200	ຂາວ, ເຫຼືອງ, ແດງ
3ຈ-7483	Misubishi	Lancer	ຂາວ, ຂຽວ, ດຳ
4ກ-8422	Toyota	Corola	ຂາວ, ເຫຼືອງ, ແດງ, ດຳ

ເມື່ອກວດສອບພົບວ່າຕາຕະລາງໃດທີ່ມີ Field ທີ່ເປັນ Multivalues ແລ້ວ ວິທີການ
ແກ້ໄຂກໍຄືການແບ່ງ Field ດັ່ງກ່າວອອກມາເປັນອີກນຶ່ງຕາຕະລາງໂດຍໃຫ້ເປັນ Field ທຳມະດາ
(Field ທຳມະດາຄື Field ທີ່ໃນນຶ່ງເຮດຂອດຈະມີພຽງຄ່າດຽວເທົ່ານັ້ນ) ແລ້ວ ດຶງເອົາ Primary
key ຂອງຕາຕະລາງຫຼັກມານຳ

ຕົວຢ່າງ (Entity ຜ່ານ 1NF)

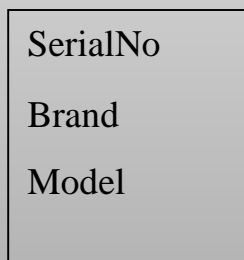
Table: Car

SerialNo	Brand	Model
1ຂ-2778	Benz	E200
3ຈ-7483	Misubishi	Lancer
4ກ-8422	Toyota	Corola

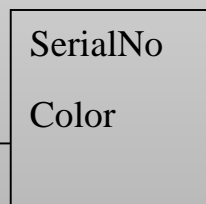
Table: CarColor

SerialNo	Color
1ຂ-2778	ຂາວ
1ຂ-2778	ເຫຼືອງ
1ຂ-2778	ແດງ
3ຈ-7483	ຂາວ
3ຈ-7483	ຊຽວ
3ຈ-7483	ດຳ
4ກ-8422	ຂາວ
4ກ-8422	ເຫຼືອງ
4ກ-8422	ແດງ
4ກ-8422	ດຳ

Car



CarColor



2NF (Second Normal Form)

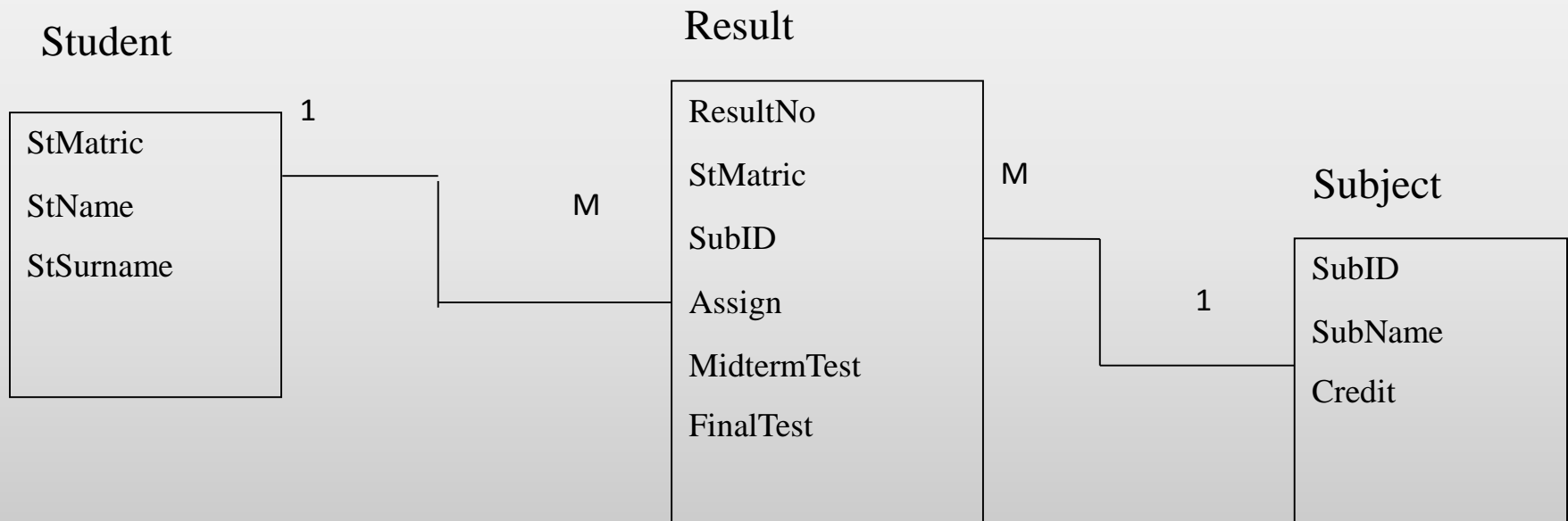
Entity ໃດທີ່ຜ່ານ 2NF ນັ້ນຈະຕ້ອງຜ່ານ 1NF
ແລະ Entity ນັ້ນ ຕ້ອງມີພຽງແຕ່ໜຶ່ງ Attribute ທີ່ມີ
ຄຸນສົມບັດເປັນ Primary key

ຕົວຢ່າງ: Entity: Result ຍັງບໍ່ທັນຜ່ານການ Normalization

Entity: Result

Result No	SubID	StMatric	StName	StSurname	SubName	PhoneNo	Credit
1	SC01	St0001	Kham	Phongvilay	Database	770658	3
2	SC01	St0002	Keo	Boutaly	Database	770621	3
3	SC01	St0003	Vandy	Sisopha	Database	770896	3
4	SC02	St0001	Kham	Phongvilay	OS	770658	3
5	SC02	St0002	Keo	Boutaly	OS	770621	3
6	SC03	St0001	Kham	Phongvilay	OOP		3

ຕົວຢ່າງ: Entity: Result ຜ່ານ 2NF



ຕົວຢ່າງ: Entity: Result ຜ່ານການເຮັດ Normalization

Student

StMatric	StNname	StSurname	PhoneNO
St0001	Kham	Phongvilay	770658
St0002	Keo	Boutaly	770621
St0003	Vandy	Sisopha	770896

Subject

SubID	SubName	Credit
SC01	Database	3
SC02	Programming1	3
SC03	Programming2	3

Result

ResultNo	SubID	StMatric	Assign	MidTest	FinalTest
1	SC01	St0001	15	25	40
2	SC01	St0002	19	20	45
3	SC01	St0003	16	29	30
4	SC02	St0001	18	27	46

3FN (Third Normal Form)

Entity ໃດທີ່ຜ່ານ 3NF ນັ້ນຈະຕ້ອງຜ່ານ 1NF, 2NF ແລະ ຈະຕ້ອງບໍ່ມີ Attribute ໃດໃນ Entity ນອກ ຈາກ Primary Key ສາມາດກຳນົດ Attribute ອື່ນໄດ້

ຕົວຢ່າງ: Entity: InvoiceDetail ຍັງບໍ່ຜ່ານ 3NF

Entity: InvoiceDetail

Invoice No	Product No	Amount	UnitPrice	CustID	CustName	PhoneNo	Address
10001	P0001	25	15000	CU0001	Mr. Khamla	770632	21 Nongbone
10001	P0002	10	50000	CU0001	Mr. Khamla	770632	
10001	P0003	60	45000	CU0001	Mr. Khamla	770632	
10002	P0005	25	12500	CU0002	Mis. Vanhdy	614432	15 Phonekheng
10002	P0001	50	31500	CU0002	Ms. Vanhdy	614432	
10002	P0003	20	14500	CU0002	Ms. Vanhdy	614432	

ផ្ទាំង 3NF

ព្រឹត្តិការណ៍ Invoice

InvoiceNo	ProductNo	Amount	UnitPrice	CustID
10001	P0001	25	15000	CU0001
10001	P0002	10	50000	CU0001
10001	P0003	60	45000	CU0001
10002	P0005	25	12500	CU0002
10002	P0001	50	31500	CU0002
10002	P0003	20	14500	CU0002

ព្រឹត្តិការណ៍ Customer

CustID	CustName	PhoneNo	Address
CU0001	Mr. Khamla	770632	21 Nongbone
CU0001	Mr. Khamla	770632	
CU0001	Mr. Khamla	770632	
CU0002	Mis. Vanhdy	614432	15 Phonekheng Village
CU0002	Ms. Vanhdy	614432	
CU0002	Ms. Vanhdy	614432	

ສະຫຼຸບ

ສໍາລັບການເຮັດ Normalization ນີ້ສ່ວນໃຫຍ່ເມື່ອເຖິງລະດັບ 3NF ກໍສາມາດກໍາຈັດບັນຫາທີ່ເກີດຂຶ້ນໄດ້ທັງໝົດ, ສ່ວນທີ່ຕ້ອງເຮັດຕໍ່ໄປຈົນເຖິງລະດັບ 4NF ແລະ 5NF ນັ້ນຈະບໍ່ຄ່ອຍພົບເນື່ອງຈາກໃນທາງປະຕິບັດຕົວຈິງບາງກໍລະນີນັ້ນອາດຈໍາເປັນຕ້ອງຍອມໃຫ້ມີຂໍ້ມູນຊໍ້າກັນໄດ້ເພື່ອໃຫ້ການຄົ້ນຫາ ຫຼື ປະມວນຜົນເປັນໄປໄດ້ຢ່າງວ່ອງໄວ, ເພາະວ່າຖ້າເຮັດ Normal Form ຫຼາຍເກີນໄປຈະເຮັດໃຫ້ໂຄງສ້າງຖານຂໍ້ມູນຊັບຊ້ອນ

Thank you

Q and A