ICA | INFORMATICA COMMUNICATIE ACADEMIE

Building an ER-model from verbalizations of facts

Part 2: Analyzing FTs with 2 segments

HAN

Procedure to draw up an ERD

Steps 1, 2 and 3: not covered here.

Step 1: see the reader DM-RDS.

Step 2: see presentation Verbalizing.

Step 3: see presentation Sorting.

Step 4:

this presentation.

1. Collect concrete examples of facts
   - Use BPM as starting point
   - Make up examples if they don't exist (yet)
2. Verbalize these examples
   - With domain expert. Result: fact expressions.
   - Make the meaning as clear as possible
3. Sort expressions into Fact Types (FTs)
   - Same kind of expression: same FT
   - Order FTs with most components last
4. Analyze each FT (2 segments) and add the results to the ERD

Sorting fact expressions

Expressions of the same kind belong to a **Fact Type**.

Expressions of the same type have **components**: places where the text can vary.

FT4 has 2 components

FT6 has 3 components

```
FT4
Employee InsEd manages project P315.
    "    SmthE    "       "    P422.

FT6
Subproject 1 of project P315 starts on 20160201.
    "      2  "    "    P315    "    " 20160201.
    "      3  "    "    P315    "    " 20160208.
    "      1  "    "    P422    "    " 20160101.
```

Analyzing fact types

No matter how many components a FT has,
it can have only 1 or 2 **segments:**
groups of components that belong together.

Analzying fact types is:

determining which segments there are,
and which ETs, Atts and RTs are involved.

Analyzing fact types

The cases with 2 segments are treated in this presentation.

For a FT with 2 segments there are only 2 possibilities:

- One segment concerns an ET,
  the other segment concerns an Att of this ET.
- Both segments concern ETs,
  then there is also a RT that connects these ETs.

The cases with 1 segment are treated in presentation 3_6.

There is only one possibility for a FT with 1 segment:

- The segment concerns an ET

Analyzing fact types

The procedure to analyze FTs will be illustrated using the following four FTs:

```
FT1:
The family name of student S17 is Johansen.
 "     "     "   "    "    T66  " Robberts.

FT2:
The course SQL is taught by Tmina.
 "     "   ERM  "    "    " Ttigo.

FT3:
The exam for the course SQL on 14/1/2016 is held in room R67.
 "    "   "   "     "   ERM  " 25/2/2016  "   "   "   "  45a.

FT4:
Student T66 scored a mark of 85 for the exam of SQL on 14/1/2016.
   "    S17   "    "   "   " 47  "   "    "   " ERM  " 25/2/2016.
```

All modeling decisions were discussed with domain experts.

Students must do all analyses <u>exactly</u> as shown in the next slides.

Analyzing fact types: FT1 (ET+Att)

Two components:

- Underline segments

```
FT1:
The family name of student S17 is Johansen.
 "     "     "   "    "    T66  " Robberts.
```

Analyzing fact types: FT1 (ET+Att)

Two components:

- Underline segments
- Indicate type of segment: one is ET, one is Att.

```
FT1:
The family name of student S17 is Johansen.
 "     "     "   "    "    T66  " Robberts.
                    ET            Att
```

Analyzing fact types: FT1 (ET+Att)

Two components:

- Underline segments
- Indicate type of segment: one is ET, one is Att.
- Give them meaningful names

FT1:
The family name of student S17 is Johansen.
" " " " " T66 " Robberts.
ET Att

Analyzing fact types: FT1 (ET+Att)

Two components:

- Underline segments
- Indicate type of segment: one is ET, one is Att.
- Give them meaningful names

FT1:
The family name of student S17 is Johansen.
" " " " " T66 " Robberts.
ET STUDENT Att Family_name

Analyzing fact types: FT1 (ET+Att)

Identifier of STUDENT: S17 and T66 are student numbers, which are called ‘Studno’ according to the domain expert.

For each ET: establish its <pi> (if Att: always <M>)

```
FT1
The family name of student S17 is Johansen.
 "     "     "   "    "    T66   " Robberts.
                ET STUDENT      Att Family_name
                ID: Att Studno
```

Predicate: **The family name of student <Studno> is <Family_name>.**

ERD
The <pi> and <M> were checked with the domain experts. Domains for the Atts were specified also.

STUDENT			
Studno	<pi>	STUDNO	<M>
Family_name		NAME	<M>

Rules for analyzing FTs

- Mark 2 segments (or 1), and decide on
  ET + Att or ET + ET (if 1 segment: ET).
-
- If you find a new ET:
  determine its ID (primary identifier)
-
-
- Give the complete predicate
- Determine <M> for new Atts
-
-

Analyzing fact types: FT2 (ET+ET)

Two components:

- Underline segments
- Indicate type of segment: here, both are ET
- Give them meaningful names
- Determine primary identifier

FT2:
The course SQL is taught by Tmina.
" " ERM " " " Ttigo.
ET COURSE ET TEACHER
ID: Att Course_code ID: Att Teacher_code

Analyzing fact types: FT2 (ET+ET)

Add a RT between the ETs; determine its cardinalities

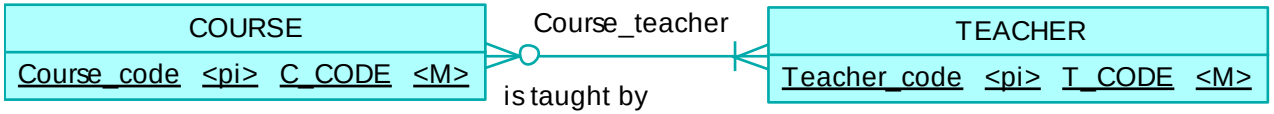
```
FT2
The course SQL is taught by Tmina.
 "      "    ERM   "     "     " Ttigo.
ET COURSE                      ET TEACHER
ID: Att Course_code            ID: Att Teacher_code
```

RT: explicit notation with ET-names needed in general

```
RT Course_teacher between COURSE and TEACHER

Predicate: The course <Course_code>
           is taught by <Teacher_code>.
```

ERD
All constraints, domains and cardinalities were determined with the domain experts



Rules for analyzing FTs

- Mark 2 segments (or 1), and decide on
  ET + Att or ET + ET (if 1 segment: ET).
-
- If you find a new ET:
  determine its ID (primary identifier)
-
- In the ET + ET case:
  add a **non-dependent RT**
- Give the complete predicate
- Determine <M> for new Atts
- Determine cardinalities for new RTs
-

Analyzing fact types: FT3 (weak ET)

Three components:

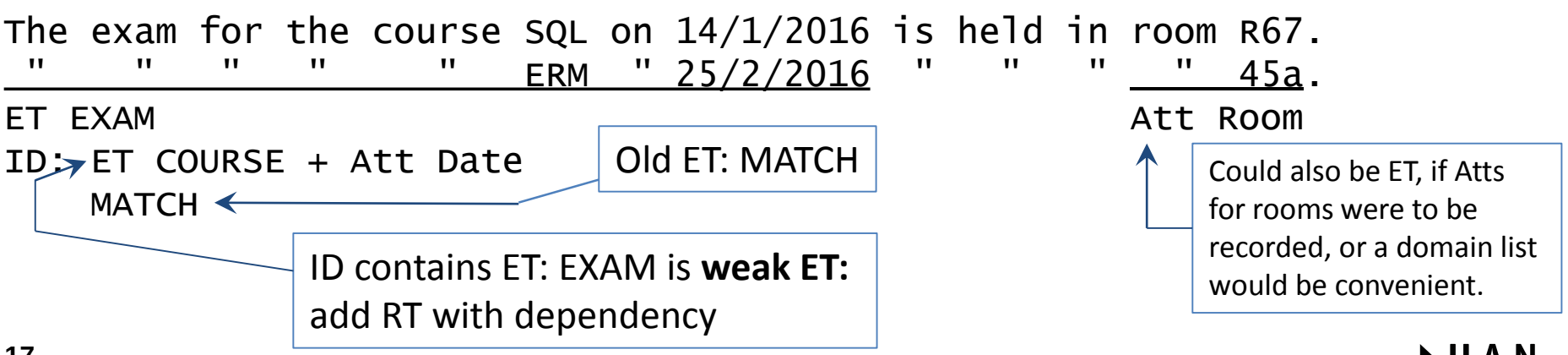
- Underline segments
  - 3 components, 2 segments: combine 2 components in one segment
  - Which combination of components identifies something meaningful?
  - Here: ‘the exam for the course SQL on 14/1/2016’ identifies an Exam

```
FT3:
The exam for the course SQL on 14/1/2016 is held in room R67.
 "    "   "   "     "   ERM  " 25/2/2016  "   "   "   "  45a.
```

Analyzing fact types: FT3 (weak ET)

Three components:

- Underline segments
  - 3 components, 2 segments: combine 2 components in one segment
  - Which combination of components identifies something meaningful?
  - Here: 'the exam for the course SQL on 14/1/2016' identifies an Exam
- Indicate type of segment and give them meaningful names: ET Exam + Att Room
- Determine primary identifier for ET Exam (can be Att+Att, ET+Att, ET+ET)
  - Here: ET + Att



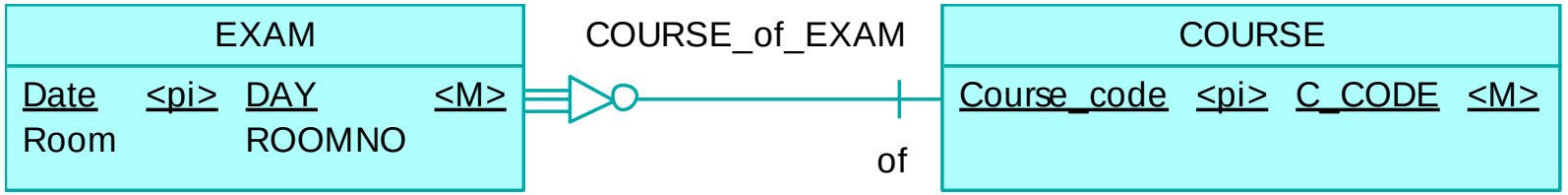
Analyzing fact types: FT3 (weak ET)

FT3:
The exam for the course SQL on 14/1/2016 is held in room R67.
" " " " " ERM " 25/2/2016 " " " " 45a.
ET EXAM Att Room
ID: ET COURSE + Att Date
MATCH

ID contains ET: EXAM is **weak ET:** add RT with dependency

RT COURSE_of_EXAM between EXAM(dependent)
and COURSE

Predicate: **The exam for the course <Course_code> on <Date> is held in room <Room>.**

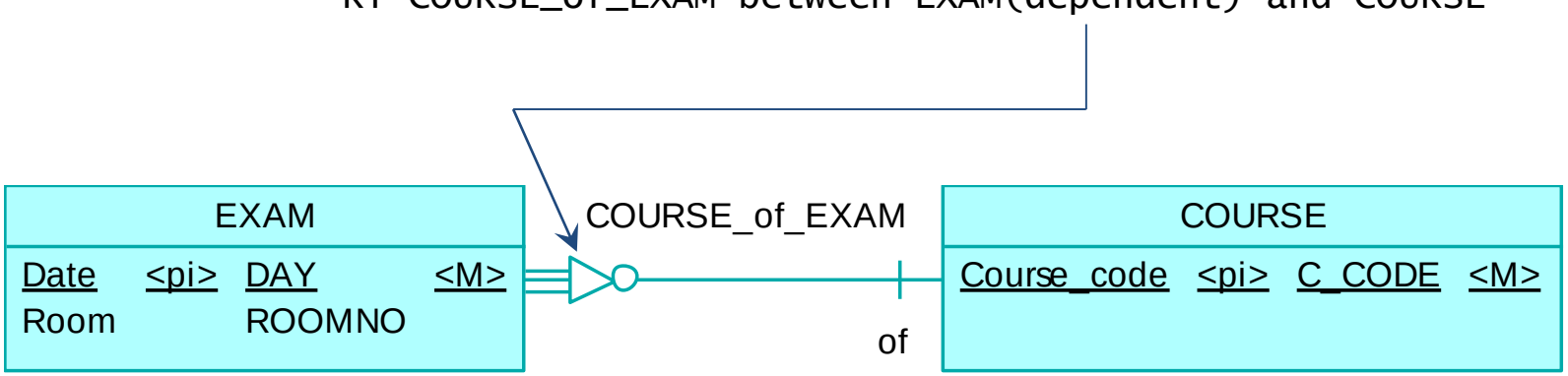


Weak Entity Type

When an ET is **identified** by **one or more other ET's** it is a **weak ET**

For each ET in the identifier add a dependent RT:

```
RT COURSE_of_EXAM between EXAM(dependent) and COURSE
```



Rules for analyzing FTs

- Mark 2 segments (or 1), and decide on
  ET + Att or ET + ET (if 1 segment: ET).
- If you find an old ET: MATCH
- If you find a new ET:
  determine its ID (primary identifier)
- If this ID contains an ET:
  add a **dependent RT** to it
- In the ET + ET case:
  add a **non-dependent RT**
- Give the complete predicate
- Determine <M> for new Atts
- Determine cardinalities for new RTs
-

Analyzing fact types: FT4 (Complex)

```
FT4:
Student T66 scored a mark of 85 for the exam of SQL on 14/1/2016.
   "    S17   "    "   "   " 47  "   "    "   " ERM  " 25/2/2016.
```

How to combine these 4 components into 2 segments?

Observe:

- Two ‘old’ ETs are present: STUDENT and EXAM
  these make up 3 of the 4 components.
- The 4th component is new: a mark.
- A mark is not a property (attribute) of a student alone,
  nor is it a property of an exam alone,
  but it is a property of the combination of a student and an exam:
  the domain expert agrees this combination is called an EXAM PARTICIPATION.

So: 3 components make up an ET EXAM_PARTICIPATION, the 4th an Att Mark.

Analyzing fact types: FT4 (Complex)

```
FT4:
Student T66 scored a mark of 85 for the exam of SQL on 14/1/2016.
   "    S17   "    "   "   " 47  "   "    "   " ERM  " 25/2/2016.
                   Att Mark
```

```
ET EXAM_PARTICIPATION
ID: ET STUDENT + ET EXAM
    MATCH        MATCH
```

Old ETs STUDENT and EXAM present. Mark: attribute (property) of an exam participation. So other three components must be one ET.

ID contains 2 old ETs: 2 MATCHes

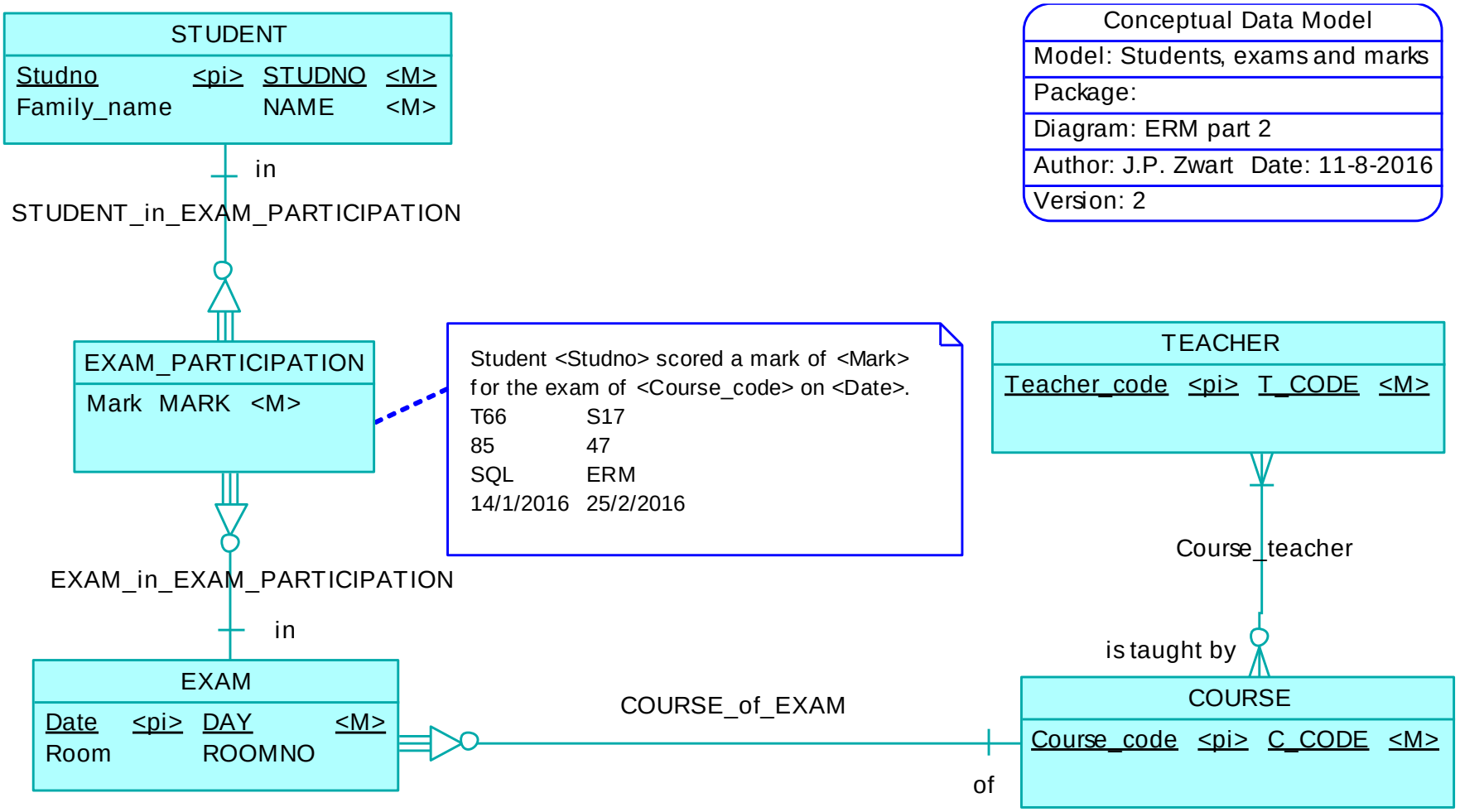
For each ET in the ID: add a dependent RT

```
RT STUDENT_in_EXAM_PARTICIPATION between
   EXAM_PARTICIPATION(dependent) and STUDENT

RT EXAM_in_EXAM_PARTICIPATION between
   EXAM_PARTICIPATION(dependent) and EXAM

Predicate: Student <Studno> scored a mark of <Mark>
           for the exam of <Course_code> on <Date>.
```

Analyzing fact types: Complete ERD



Rules for analyzing FTs

- Mark 2 segments (or 1), and decide on
  ET + Att or ET + ET (if 1 segment: ET).
- If you find an old ET: MATCH
- If you find a new ET:
  determine its ID (primary identifier)
- If this ID contains an ET:
  add a dependent RT to it
- In the ET + ET case:
  add a non-dependent RT
- Give the complete predicate
- Determine <M> for new Atts
- Determine cardinalities for new RTs
- Add predicates and populations to the diagram to make the meaning of the fact types more clear

Practical recommendations

- Always work exclusively from concrete examples of facts.
- Always verbalize these facts carefully, with the possible exception of widely known simple attributes, but don't be too sloppy!
- Add predicates and/or example populations for
  - all unclear non-dependent RTs
  - all unclear <pi>+Att fact types