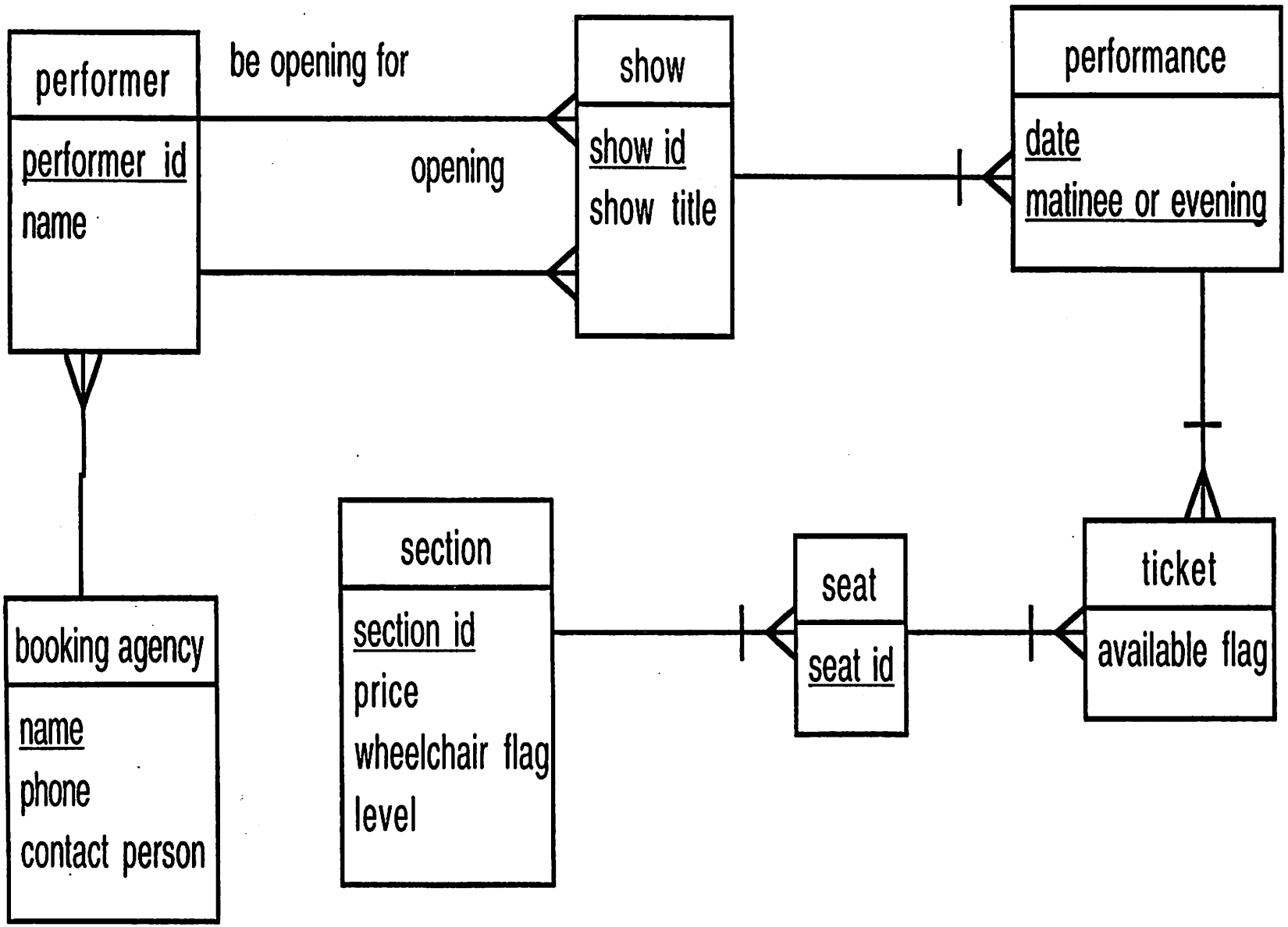
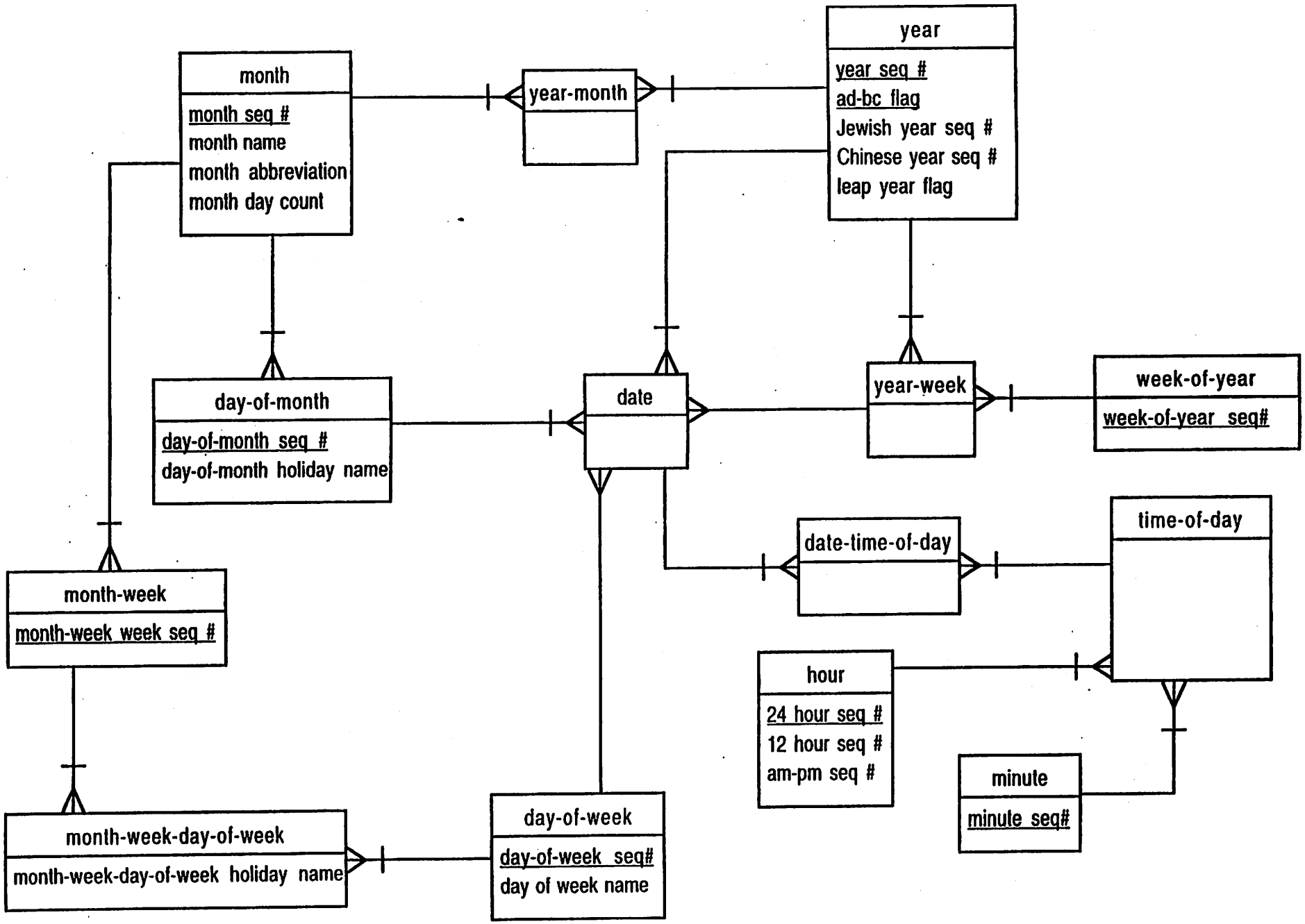

performer
performer id
name
be opening for
opening
show
show id
show title
performance
date
matinee or evening
booking agency
name
phone
contact person
section
section id
price
wheelchair flag
level
seat
seat id
ticket
available flag


month
month seq #
month name
month abbreviation
month day count
year-month
year
year seq #
ad-bc flag
Jewish year seq #
Chinese year seq #
leap year flag
day-of-month
day-of-month seq #
day-of-month holiday name
date
year-week
week-of-year
week-of-year seq#
month-week
month-week week seq #
date-time-of-day
time-of-day
hour
24 hour seq #
12 hour seq #
am-pm seq #
minute
minute seq#
month-week-day-of-week
month-week-day-of-week holiday name
day-of-week
day-of-week seq#
day of week name

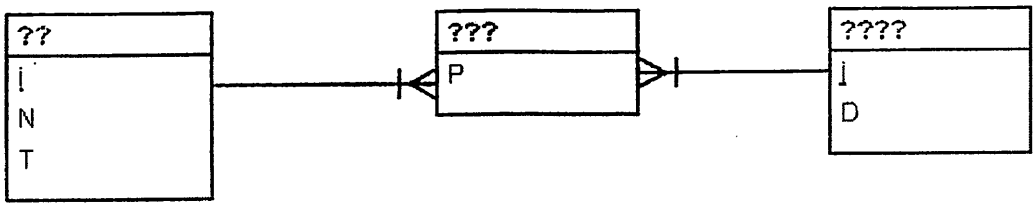
Notes about scientific notation

Suppose you are reading a document which includes this terse scientific notation:

$$N_i, \quad T_i, \quad D_j, \quad P_{ij}$$

To understand it better you can reverse engineer a data model where:

- Each subscript becomes an identifying descriptor, as appropriate, attribute or link.
  - A subscript such as "i", which is by itself with "N" maps to an identifying attribute, identifying an independent entity. Likewise for "j".
  - The "i" in P_{ij} maps to an identifying link, partially identifying an intersection or dependent entity. [In this case it's an intersection entity
- Each variable becomes a non-identifying attribute of an entity with the variable's subscript(s) serving as the entity's identifier.



To make the data model well-formed and understandable, you need to:

- Determine each entity name, which may be explicit in the document or be implicit, requiring you to pick the name
- Substitute a human-understandable name for each terse attribute.

In this case, pick entity names and make these substitutions:

- i → C_id; j → S_code;
  N → C_name; T → C_type;
  D → S_description; P → Proficiency.

Then, voila, the Creature-Achievement-Skill fragment appears.

This is a quite empowering aspect of mastering data modeling. You can understand really hard material by applying your content-neutral skills. You can do this with something you are generating or material written by others that you are reading. For example, you can find holes in notation, which can manifest as descriptor misplacement. You can ask if the subscripts, which generally have integer values, are nominal, ordinal or numeric scale, that is, is it a set or a sequence, and can one do sensible arithmetic on the values or not. You can practice the mantras "what do we mean by one of these", "one or many", and "anchor understanding with instances.

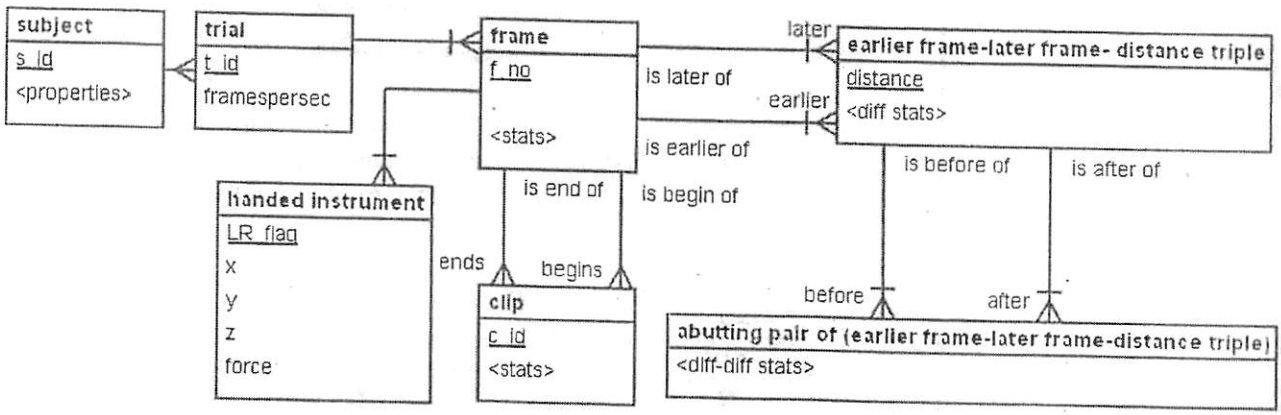
Cool, ch.

4

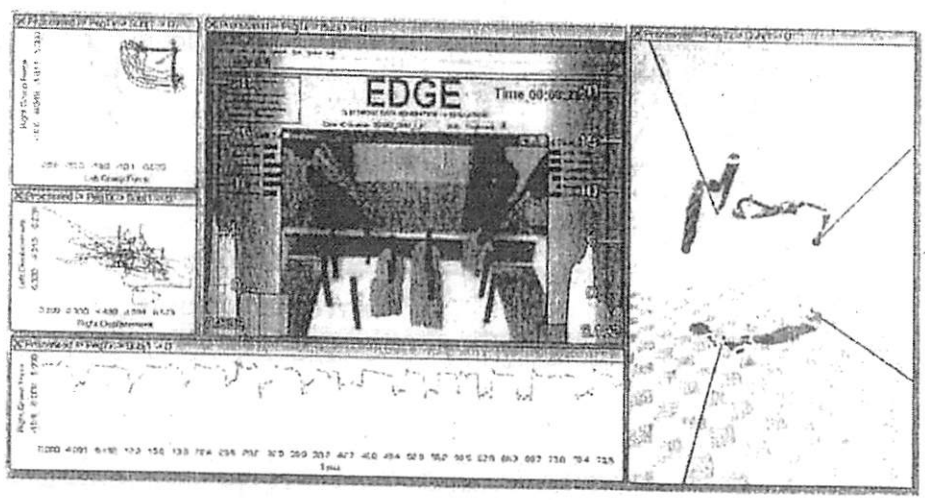
Visualizing Surgical Training Databases: Exploratory Visualization, Data Modeling, and Formative Feedback for Improving Skill Acquisition

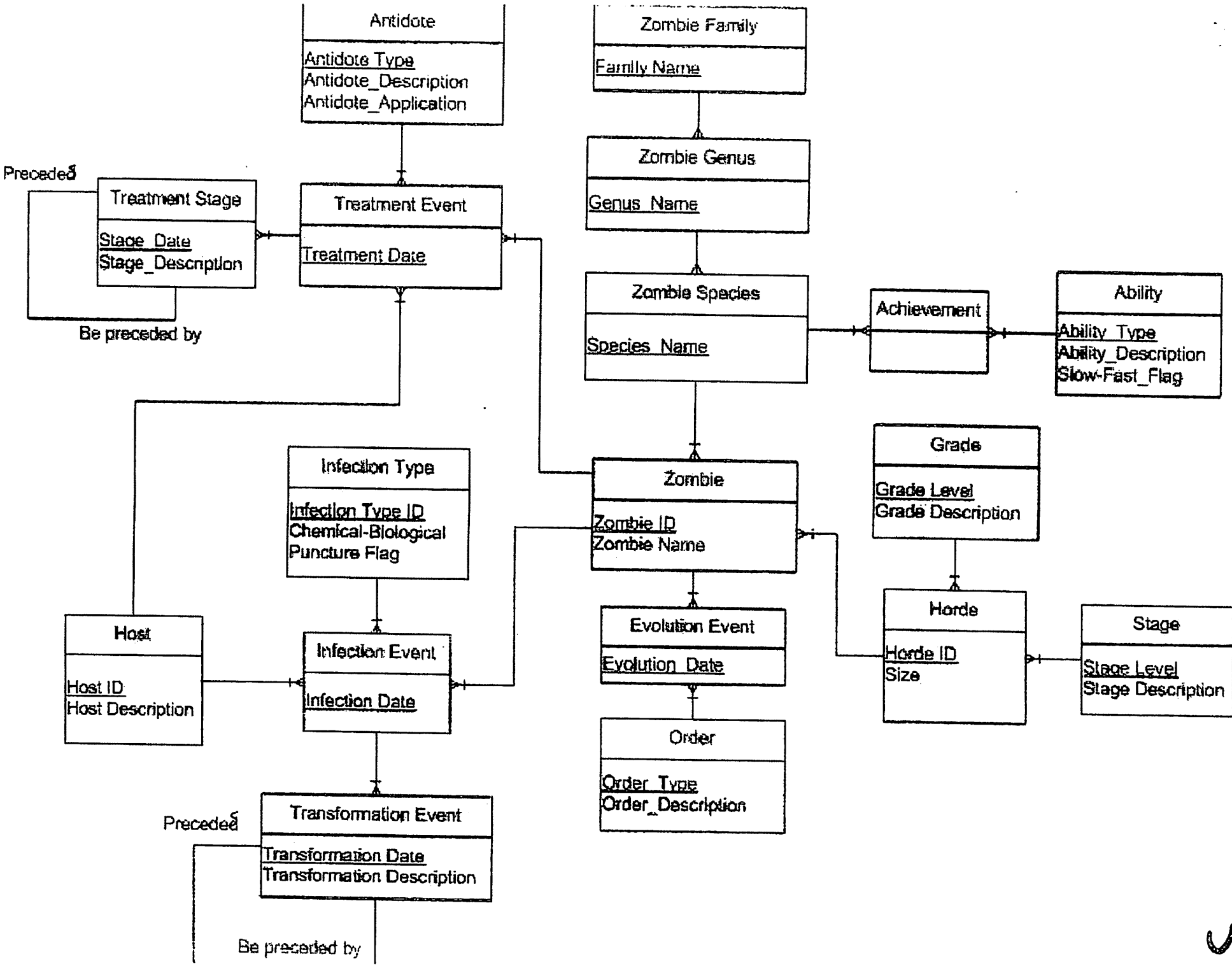
David Schroeder[1], Daniel F. Keefe[1], Timothy Kowalewski[2], Lee White[3], John Carlis[1], Erlan Santos[4], Robert Sweet[4], Thomas S. Lendvay[5], Troy Reihsen[4], Blake Hannaford[2]

DRAFT



TASK: From this data model (WHAT) form a relational schema (HOW)




Antidote
Antidote Type
Antidote_Description
Antidote_Application
Zombie Family
Family Name
Zombie Genus
Genus Name
Preceded
Treatment Stage
Stage Date
Stage_Description
Treatment Event
Treatment Date
Be preceded by
Zombie Species
Species Name
Achievement
Ability
Ability Type
Ability_Description
Slow-Fast_Flag
Grade
Grade Level
Grade Description
Infection Type
Infection Type ID
Chemical-Biological
Puncture Flag
Zombie
Zombie ID
Zombie Name
Host
Host ID
Host Description
Infection Event
Infection Date
Evolution Event
Evolution Date
Horde
Horde ID
Size
Stage
Stage Level
Stage Description
Order
Order Type
Order_Description
Preceded
Transformation Event
Transformation Date
Transformation Description
Be preceded by

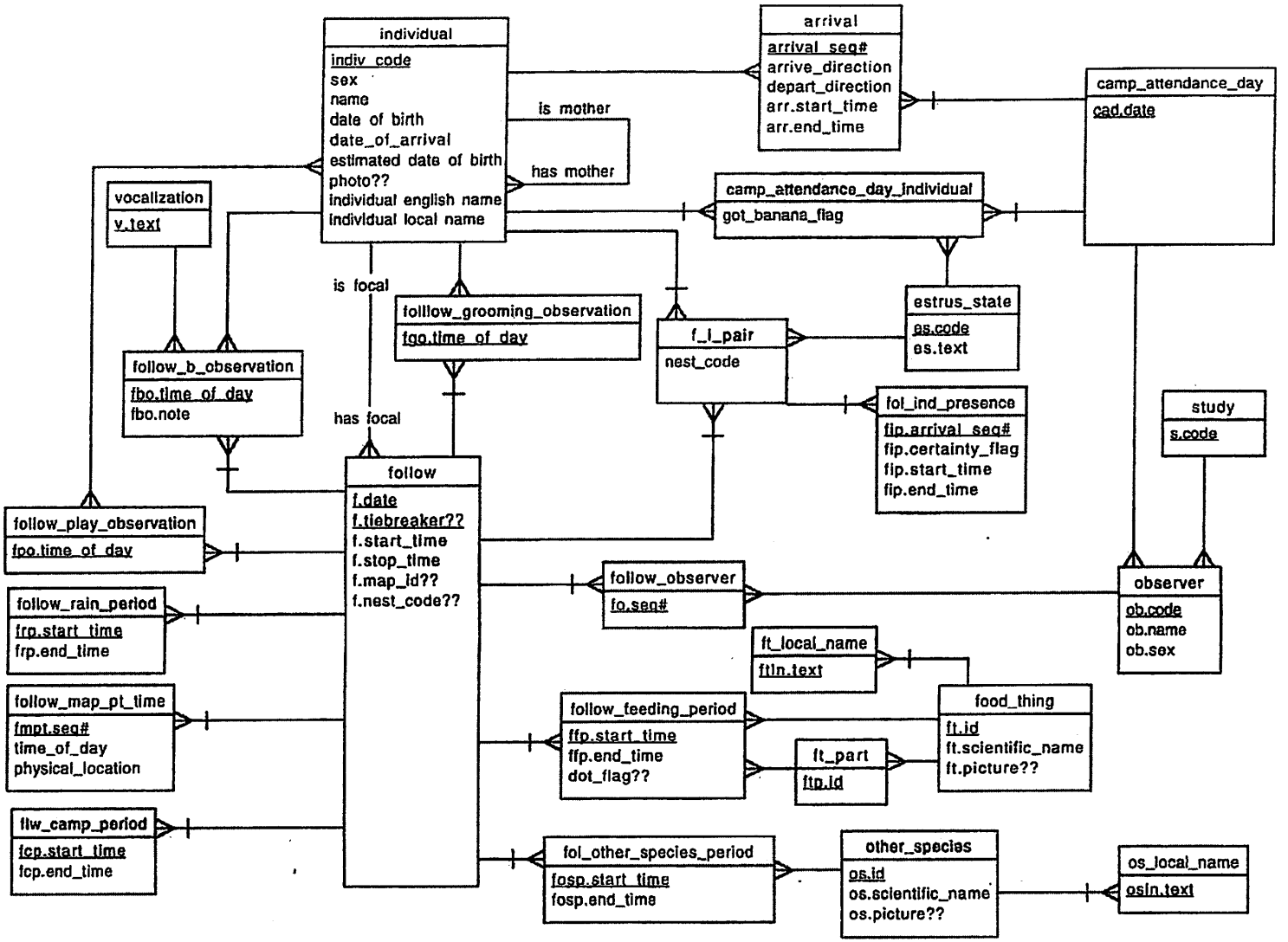

individual
indiv_code
sex
name
date of birth
date_of_arrival
estimated date of birth
photo??
individual english name
individual local name
is mother
has mother
arrival
arrival_seq#
arrive_direction
depart_direction
arr.start_time
arr.end_time
camp_attendance_day
cad.date
vocalization
v.text
camp_attendance_day_individual
got_banana_flag
is focal
has focal
follow_grooming_observation
fgo.time_of_day
f_i_pair
nest_code
estrus_state
es.code
es.text
follow_b_observation
fbo.time_of_day
fbo.note
fol_ind_presence
fip.arrival_seq#
fip.certainty_flag
fip.start_time
fip.end_time
study
s.code
follow
f.date
f.tiebreaker??
f.start_time
f.stop_time
f.map_id??
f.nest_code??
follow_play_observation
fpo.time_of_day
follow_observer
fo.seq#
observer
ob.code
ob.name
ob.sex
follow_rain_period
frp.start_time
frp.end_time
ft_local_name
ftln.text
follow_map_pt_time
fmpt.seq#
time_of_day
physical_location
follow_feeding_period
ffp.start_time
ffp.end_time
dot_flag??
food_thing
ft.id
ft.scientific_name
ft.picture??
ft_part
ftp.id
flw_camp_period
fcp.start_time
fcp.end_time
fol_other_species_period
fosp.start_time
fosp.end_time
other_species
os.id
os.scientific_name
os.picture??
os_local_name
osln.text

Figure 7.1. Roads LDS

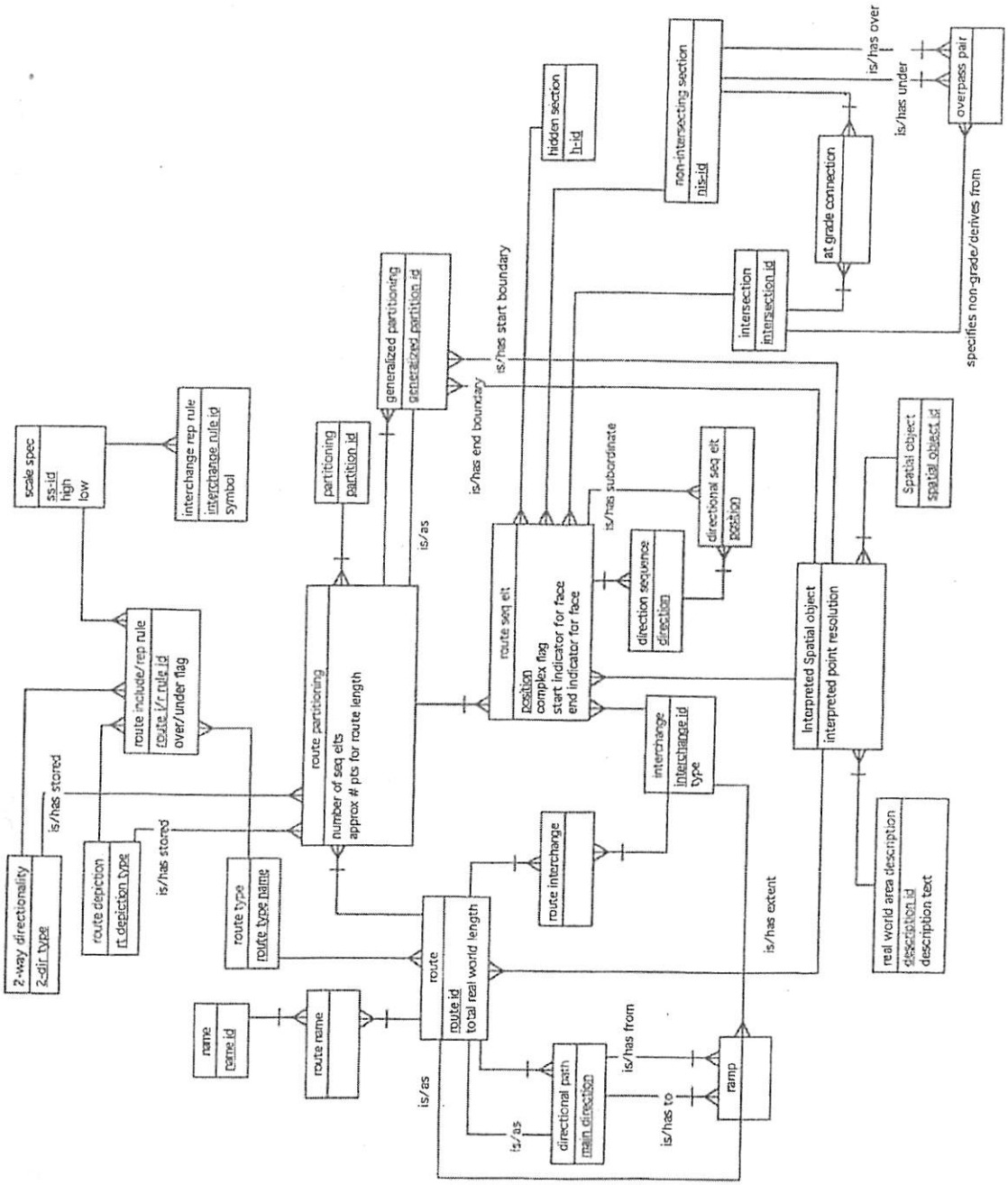
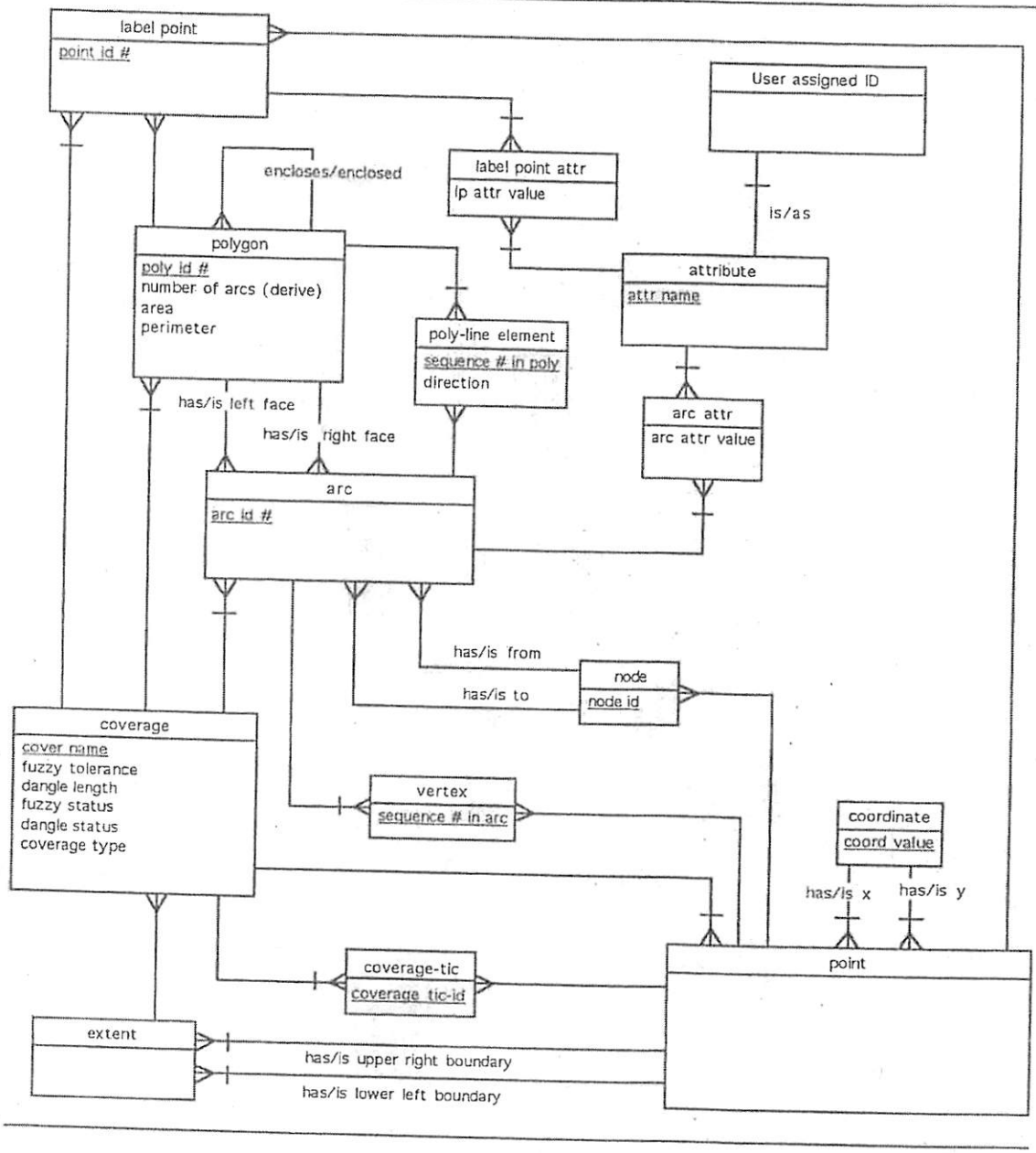


Figure 4.2. ARC/INFO LDS



a sequence of points wherein each element of the sequence is called a *vertex*. A *vertex* is like the SDTS *point* which has a "point-type" value of "entity point." An ARC/INFO *point* roughly corresponds to an SDTS "point." An ARC/INFO *polygon* has a sequence of arcs, and one element of the sequence is called a *poly-line element* in Figure 4.2. A *polygon* may have one or more interior

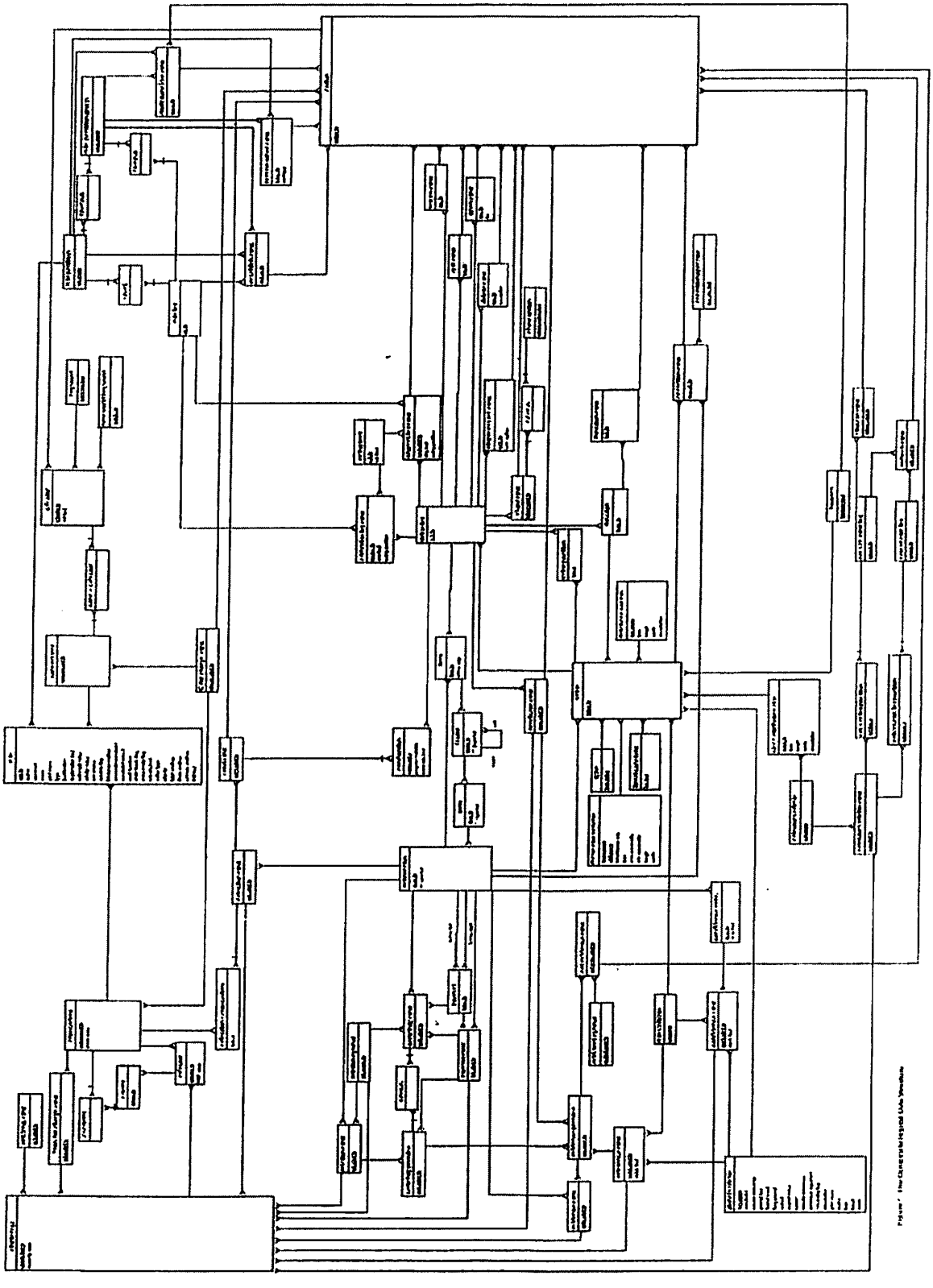
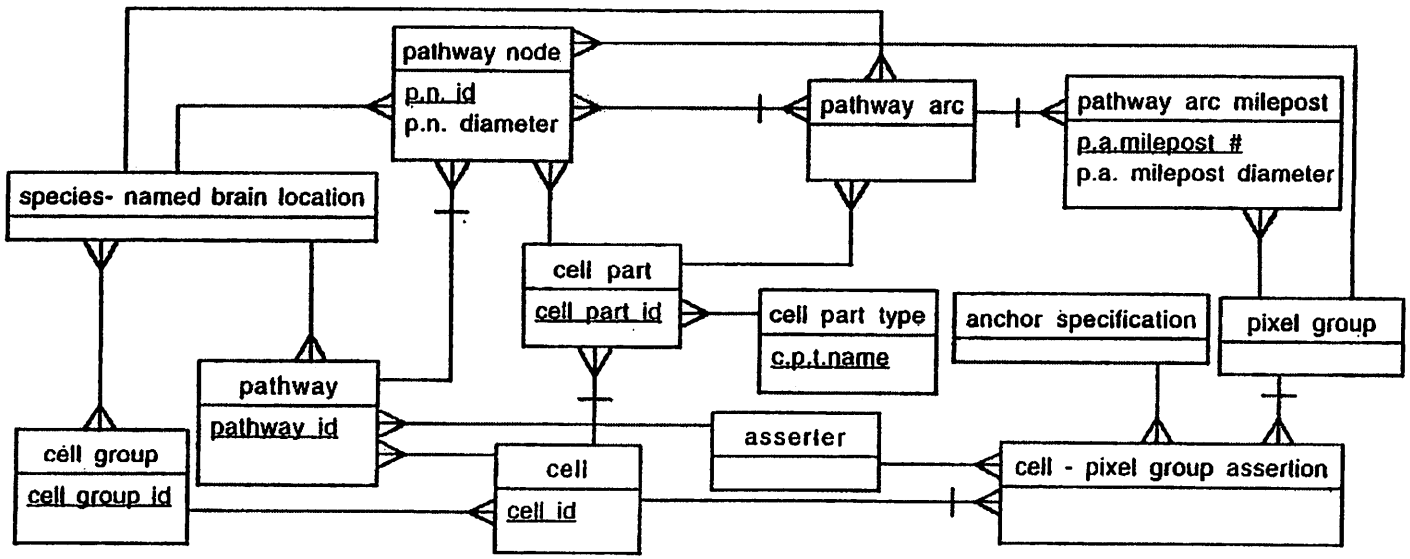
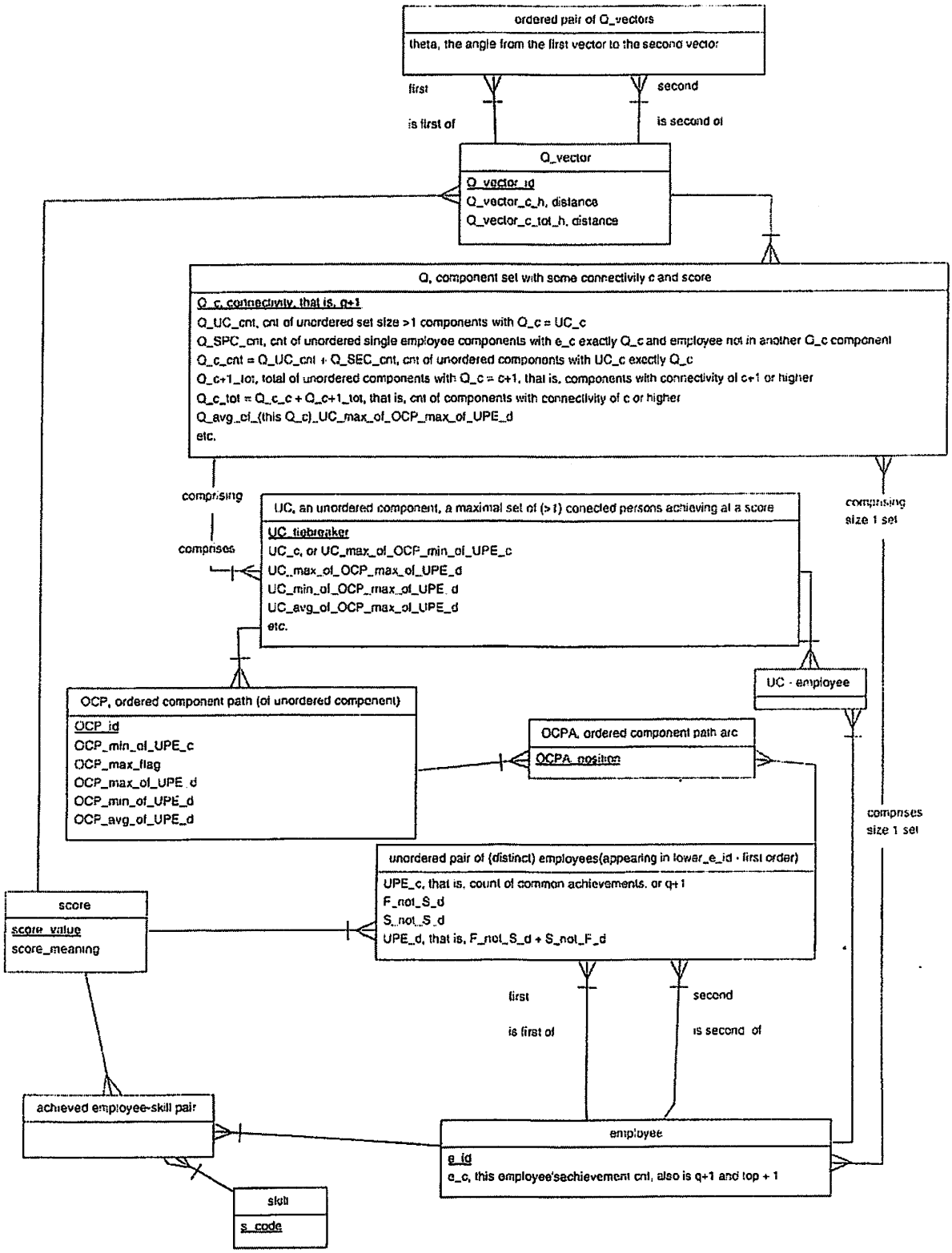


Figure 1. A modest sized LDS-


pathway node
p.n. id
p.n. diameter
pathway arc
pathway arc milepost
p.a.milepost #
p.a. milepost diameter
species- named brain location
cell part
cell part id
cell part type
c.p.t.name
anchor specification
pixel group
pathway
pathway id
asserter
cell group
cell group id
cell
cell id
cell - pixel group assertion


ordered pair of Q_vectors
theta, the angle from the first vector to the second vector
first
second
is first of
is second of
Q_vector
Q_vector_id
Q_vector_c_h, distance
Q_vector_c_tot_h, distance
Q, component set with some connectivity c and score
Q_c, connectivity, that is, q+1
Q_UC_cnt, cnt of unordered set size >1 components with Q_c = UC_c
Q_SPC_cnt, cnt of unordered single employee components with e_c exactly Q_c and employee not in another Q_c component
Q_c_cnt = Q_UC_cnt + Q_SEC_cnt, cnt of unordered components with UC_c exactly Q_c
Q_c+1_tot, total of unordered components with Q_c = c+1, that is, components with connectivity of c+1 or higher
Q_c_tot = Q_c_c + Q_c+1_tot, that is, cnt of components with connectivity of c or higher
Q_avg_of_(this Q_c)_UC_max_of_OCP_max_of_UPE_d
etc.
comprising
comprises
comprising
size 1 set
UC, an unordered component, a maximal set of (>1) conected persons achieving at a score
UC_tiebreaker
UC_c, or UC_max_of_OCP_min_of_UPE_c
UC_max_of_OCP_max_of_UPE_d
UC_min_of_OCP_max_of_UPE_d
UC_avg_of_OCP_max_of_UPE_d
etc.
UC - employee
OCP, ordered component path (of unordered component)
OCP_id
OCP_min_of_UPE_c
OCP_max_flag
OCP_max_of_UPE_d
OCP_min_of_UPE_d
OCP_avg_of_UPE_d
OCPA, ordered component path arc
OCPA_position
comprises
size 1 set
unordered pair of (distinct) employees(appearing in lower_e_id - first order)
UPE_c, that is, count of common achievements, or q+1
F_not_S_d
S_not_S_d
UPE_d, that is, F_not_S_d + S_not_F_d
score
score_value
score_meaning
first
second
is first of
is second of
achieved employee-skill pair
employee
e_id
e_c, this employee'sechievement cnt, also is q+1 and top + 1
skill
s_code

Chapter 18: Script forControlled Evolution: The Flow

Stages and Investigations