AcademicTerm			
referenced (one-to-many):			by academic_term in Program
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
start	DATE	NOT NULL	
end	DATE	NOT NULL	
start_lessons	DATE	NOT NULL	
end_lessons	DATE	NOT NULL	
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	All Programs happen in a Period. A Period is has a start and end date. No start date of a Program may be set before the periods start Program and no end date of Program may be set after the Periods end date. One can say: A period is the program of the university, where the university consists of all departments.		

Attribute			
related as object:			Person via personHasAttribute, Role via roleImpliesAttribute
timekey	TIMESTAMP	NOT NULL	DEFAULT now
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
type	STRING	NOT NULL	DEFAULT 'string'
unique_value	BOOLEAN	NOT NULL	DEFAULT false
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	The Attribute represents user-created Attributes for Persons which can be used to construct constraints.		

Building			
referenced (one-to-many):			by building in Room
id	INTEGER	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
name	STRING		
address	STRING	NOT NULL	
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Constraint	UNIQUE (address)		
Description	A Building> has a unique street address and belongs to a Department. Some Buildings have a special name ("Henry-Ford-Building") which is unique across the whole university.		

Configuration			
timekey	TIMESTAMP	NOT NULL	DEFAULT now
key	STRING	NOT NULL	
value	STRING		
Constraint	PRIMARY KEY (key)		

Course

related as subject:	CourseAttribute via courseHasCourseAttribute, Year via courseRecommendedForYear, Course via courseRequiresCourse		
related as object:	Course via courseRequiresCourse, Person via personGivesCourse, Person via personSuccessfullyPassedCourse		
referenced (one-to-many):	by part_of in CourseElement, by instance_of in CourseInstance		
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	A Course is a well-defined set of CourseElements. For example there may be a Course which consists of two CourseElements of which one is a mandatory lecture and the other one is an optional seminar.		

CourseAttribute

related as object:	Course via courseHasCourseAttribute		
timekey	TIMESTAMP	NOT NULL	DEFAULT now
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
type	STRING	NOT NULL	DEFAULT 'string'
unique_value	BOOLEAN	NOT NULL	DEFAULT false
required	BOOLEAN	NOT NULL	DEFAULT false
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	CourseAttribute represents user-created Attributes for Courses which can be used to construct constraints.		

CourseElement

related as subject:	Feature via elementRequiresFeature		
referenced (one-to-many):	by course_element in CourseElementInstance		
id	INTEGER	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
name	STRING		
part_of	FOREIGN KEY → Course	NOT NULL	
duration	INTEGER	NOT NULL	≥ 1
type	FOREIGN KEY → CourseElementType		
required	BOOLEAN	NOT NULL	DEFAULT true
Constraint	PRIMARY KEY (id)		
Description	A CourseElement is a type of event at a school or university, like a lecture or a seminar. A CourseElement has a duration and it may take place several times in a week (of which each occurrence has a certain duration). A CourseElement may have some dependencies regarding the Features the Room it is held in has. A CourseElement is part of a Course. It is of a certain type (e.g. lecture, seminar, discussion, meeting, ...).		

CourseElementInstance

related as subject:	Room via elementInstancePrefersRoom, Timeslot via elementInstancePrefersTimeslot, Feature via elementInstanceRequiresFeature, Room via elementInstanceTakesPlaceInRoom

related as object:	Person via personTakesPartInElementInstance		
referenced (one-to-many):	by element_instance in ProposedScheduling		
id	INTEGER	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
course_instance	FOREIGN KEY → CourseInstance	NOT NULL	
course_element	FOREIGN KEY → CourseElement	NOT NULL	
starting_timeslot	FOREIGN KEY → Timeslot		
duration	INTEGER	NOT NULL	≥ 1
schedulable_lesson	BOOLEAN	NOT NULL	DEFAULT true
Constraint	PRIMARY KEY (id)		
Description	A CourseElementInstance is created within a Program belonging to a CourseInstance. It is a CourseElement actually taking place.		

CourseElementType			
referenced (one-to-many):	by type in CourseElement		
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	A CourseElementType describes a type of event, like a lecture or a seminar.		

CourseInstance			
related as object:	Person via personEnrolledInCourseInstance		
referenced (one-to-many):	by course_instance in CourseElementInstance		
id	INTEGER	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
program	FOREIGN KEY → Program	NOT NULL	
instance_of	FOREIGN KEY → Course	NOT NULL	
start	DATE	NOT NULL	
end	DATE	NOT NULL	
main_lecturer	FOREIGN KEY → Person	NOT NULL	
Constraint	PRIMARY KEY (id)		
Description	A CourseInstance is created within a Program. It is a Course actually taking place. There can be several CourseInstances of the same Course within a program. There is a main lecturer being primary responsible for any CourseInstance.		

Day			
referenced (one-to-many):	by day in Timeslot		
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	A Day holds a set of Timeslots.		

Department			

referenced (one-to-many):	by superordinate_department in Department, by department in Program		
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
superordinate_department	FOREIGN KEY → Department		
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	A logical group of several entities. Persons, Buildings and Courses belong to a Department. Buildings may belong to several Departments.		

Feature			
related as object:	CourseElementInstance via elementInstanceRequiresFeature, CourseElement via elementRequiresFeature, Room via roomProvidesFeature		
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	Rooms do have certain Features which may have a certain quantity. A Feature is identified by a name ("seats", "beamers", ...). For example any Room which is suitable to hold a lecture or a seminar in it will have a certain amount of seats, a lab has a certain amount of workplaces. While an auditorium may have 300 seats suitable for doing a lecture it might have only 75 seats suitable for doing an examination.		

Group			
related as object:	Person via personBelongsToGroup		
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	Persons may be organized into different Groups, e.g. Students may belong to certain Classes. Groups may be attached to a certain Year. A Group is identified by a name. For some universities it might be desirable to define certain kinds of groups sharing certain special properties.		

Person			
related as subject:	Group via personBelongsToGroup, CourseInstance via personEnrolledInCourseInstance, Course via personGivesCourse, Attribute via personHasAttribute, Privilege via personHasPrivilege, Role via personHasRole, Timeslot via personPrefersTimeslot, Course via personSuccessfullyPassedCourse, CourseElementInstance via personTakesPartInElementInstance		
referenced (one-to-many):	by main_lecturer in CourseInstance, by created_by in Person, by modified_by in Person, by deleted_by in Person, by program_manager in Program, by created_by in ProposedScheduling, by modified_by in ProposedScheduling, by editor in courseHasCourseAttribute		
id	INTEGER	NOT NULL	
ctime	TIMESTAMP	NOT NULL	DEFAULT now
mtime	TIMESTAMP	NOT NULL	DEFAULT now
created_by	FOREIGN KEY → Person		

modified_by	FOREIGN KEY → Person		
first_name	STRING	NOT NULL	
additional_names	STRING		
last_name	STRING	NOT NULL	
email_address	STRING		
login_name	STRING	NOT NULL	
login_password	PASSWORD	NOT NULL	
is_superuser	BOOLEAN	NOT NULL	DEFAULT false
deleted	TIMESTAMP		
deleted_by	FOREIGN KEY → Person		
working_hours	INTEGER		
student_id	STRING		
staff_id	STRING		
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (email_address)		
Constraint	UNIQUE (login_name)		
Constraint	UNIQUE (student_id)		
Constraint	UNIQUE (staff_id)		
Description	A Person is a natural person who is able to login to myCourses. Therefore she is having login credentials (e.g. username and password or mail address and password). Every Person has a first name and a last name. There are many different roles at a university and there are persons who act in more than one role (e.g. there are students doing lectures, while not being professors). Every person needs to have a unique identifier regardless of the roles they belong to for administration reasons. Persons may belong to several Groups.		

Privilege			
related as object:	Person via personHasPrivilege, Role via roleImpliesPrivilege		
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	A privilege represents the permission to do specific tasks, such as logging in, changing data of courses or publishing the shedule.		

Program			
referenced (one-to-many):	by program in CourseInstance		
id	INTEGER	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
academic_term	FOREIGN KEY → AcademicTerm	NOT NULL	
department	FOREIGN KEY → Department	NOT NULL	
freezed	BOOLEAN	NOT NULL	DEFAULT false
published	BOOLEAN	NOT NULL	DEFAULT false
program_manager	FOREIGN KEY → Person	NOT NULL	
Constraint	PRIMARY KEY (id)		
Description	A Program is a set of CourseInstances. It has a running period (start and end date). A program manager is responsible for a Program. A Program belongs to a Department and to one Department only.		

ProposedScheduling			
id	INTEGER	NOT NULL	
priority	INTEGER	NOT NULL	0 ≤ priority ≤ 100
ctime	TIMESTAMP	NOT NULL	DEFAULT now
mtime	TIMESTAMP	NOT NULL	DEFAULT now
created_by	FOREIGN KEY → Person		
modified_by	FOREIGN KEY → Person		
element_instance	FOREIGN KEY → CourseElementInstance	NOT NULL	
timeslot	FOREIGN KEY → Timeslot	NOT NULL	
room	FOREIGN KEY → Room	NOT NULL	
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (element_instance, timeslot, room)		

Role			
related as subject:	Attribute via roleImpliesAttribute, Privilege via roleImpliesPrivilege		
related as object:	Person via personHasRole		
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name)		
Description	A role bundles a set of privileges i.e. permissions a user gains when having one or more roles. The right managment is not hierarchal, it is plain.		

Room			
related as subject:	Timeslot via roomPrefersTimeslot, Feature via roomProvidesFeature		
related as object:	CourseElementInstance via elementInstancePrefersRoom, CourseElementInstance via elementInstanceTakesPlaceInRoom		
referenced (one-to-many):	by room in ProposedScheduling		
id	INTEGER	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
name	STRING		
number	STRING	NOT NULL	
building	FOREIGN KEY → Building	NOT NULL	
Constraint	PRIMARY KEY (id)		
Constraint	UNIQUE (name, building)		
Description	A Room is a physical location where CourseElementInstances take place. A Room has an identifier which uniquely identifies it across the whole university. A Room is located inside a Building. Every Room has a number which is unique in the Building. A Room has Features and it may have a certain quantity of any Feature (e.g. 300 seats, a beamer, 20 workstations). Some Rooms do have a special name ("Audimax") which is unique in the building. A Room may be of a certain type (auditorium, lab, ...). Not all rooms are always available.		

Timeslot		
related as object:	CourseElementInstance via elementInstancePrefersTimeslot, Person via personPrefersTimeslot, Room via roomPrefersTimeslot	
referenced (one-to-many):	by starting_timeslot in CourseElementInstance, by timeslot in ProposedScheduling, by timeslot in elementInstanceTakesPlaceInRoom, by timeslot in personTakesPartInElementInstance	
id	INTEGER	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
day	FOREIGN KEY → Day	NOT NULL	
startingTime	TIME	NOT NULL	
Constraint	PRIMARY KEY (id)	
Description	A Timeslot represents a duration in time in which a CourseElementInstance can take place.	

Year		
related as object:	Course via courseRecommendedForYear	
id	INTEGER	NOT NULL	
name	STRING	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (id)	
Constraint	UNIQUE (name)	
Title	year	
Description	Typically Students start studying sometime and end studying sometime later and their course of studies has a regular length. Of course Courses a Student has to enroll in in the same year must not overlap. That is what Year is for. Courses may be assigned to a certain Year (e.g. first year, second year), but they do not necessarily have to.	

courseHasCourseAttribute (Course → CourseAttribute)		
course	FOREIGN KEY → Course	NOT NULL	
attribute	FOREIGN KEY → CourseAttribute	NOT NULL	
time	TIMESTAMP	NOT NULL	DEFAULT now
editor	FOREIGN KEY → Person	NOT NULL	
value	STRING		
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (course, attribute, time)	
Description	This relation tells us that the given Course has the given Attribute.	

courseRecommendedForYear (Course → Year)		
course	FOREIGN KEY → Course	NOT NULL	
year	FOREIGN KEY → Year	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (course, year)	
Description	This relation gives the Person(students) a hint in which Course they should enroll in order to maintain their studies in order.	

courseRequiresCourse (Course → Course)		
course	FOREIGN KEY → Course	NOT NULL	

dependency	FOREIGN KEY → Course	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (course, dependency)		
Description	This relation tells us that it is neccessary to take part in the first Course in order to take part in the second Course.		

elementInstancePrefersRoom (CourseElementInstance → Room)			
element_instance	FOREIGN KEY → CourseElementInstance	NOT NULL	
room	FOREIGN KEY → Room	NOT NULL	
priority	INTEGER	NOT NULL	0 ≤ priority ≤ 100
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (element_instance, room)		
Description	This relation describes that the given CourseElementInstance is better to be placed in the given Room.		

elementInstancePrefersTimeslot (CourseElementInstance → Timeslot)			
room	FOREIGN KEY → CourseElementInstance	NOT NULL	
timeslot	FOREIGN KEY → Timeslot	NOT NULL	
priority	INTEGER	NOT NULL	0 ≤ priority ≤ 100
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (room, timeslot)		
Description	This relation describes that the given CourseElementInstance is better to be placed in the given Timeslot.		

elementInstanceRequiresFeature (CourseElementInstance → Feature)			
element_instance	FOREIGN KEY → CourseElementInstance	NOT NULL	
feature	FOREIGN KEY → Feature	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
priority	INTEGER	NOT NULL	0 ≤ priority ≤ 100
quantity_min	INTEGER		
quantity_better	INTEGER		
Constraint	PRIMARY KEY (element_instance, feature)		
Description	This relation tells us what kind of Feature is needed or whished by the Person which holds this particular CourseElementInstance.		

elementInstanceTakesPlaceInRoom (CourseElementInstance → Room)			
element_instance	FOREIGN KEY → CourseElementInstance	NOT NULL	
room	FOREIGN KEY → Room	NOT	

room	FOREIGN KEY → Room	NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
timeslot	FOREIGN KEY → Timeslot	NOT NULL	
Constraint	PRIMARY KEY (element_instance, room)		
Constraint	UNIQUE (room, timeslot)		
Description	This relation tells us that the mentioned CourseElementInstance is taking place in the given Room.		

elementRequiresFeature (CourseElement → Feature)

course_element	FOREIGN KEY → CourseElement	NOT NULL	
feature	FOREIGN KEY → Feature	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
priority	INTEGER	NOT NULL	$0 \leq priority \leq 100$
quantity_min	INTEGER		
quantity_better	INTEGER		
Constraint	PRIMARY KEY (course_element, feature)		
Description	This relation tells us what kind and amount of Feature is required in order to be held.		

personBelongsToGroup (Person → Group)

user	FOREIGN KEY → Person	NOT NULL	
group	FOREIGN KEY → Group	NOT NULL	
ctime	TIMESTAMP	NOT NULL	DEFAULT now
mtime	TIMESTAMP	NOT NULL	DEFAULT now
created_by	FOREIGN KEY → Person		
modified_by	FOREIGN KEY → Person		
Constraint	PRIMARY KEY (user, group)		
Description	This relation tells us that the given Person is part of the mentioned Group.		

personEnrolledInCourseInstance (Person → CourseInstance)

user	FOREIGN KEY → Person	NOT NULL	
course_instance	FOREIGN KEY → CourseInstance	NOT NULL	
ctime	TIMESTAMP	NOT NULL	DEFAULT now
mtime	TIMESTAMP	NOT NULL	DEFAULT now
created_by	FOREIGN KEY → Person		
modified_by	FOREIGN KEY → Person		
Constraint	PRIMARY KEY (user, course_instance)		
Description	This relation gives us the Persons that are enrolled to a specific CourseInstance.		

personGivesCourse (Person → Course)

person	FOREIGN KEY → Person	NOT NULL	
course	FOREIGN KEY → Course	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
priority	INTEGER	NOT NULL	$0 \leq priority \leq 100$
Constraint	PRIMARY KEY (person, course)		

Description	This relation tells us which Persons are holding which Courses.

personHasAttribute (Person → Attribute)			
user	FOREIGN KEY → Person	NOT NULL	
attribute	FOREIGN KEY → Attribute	NOT NULL	
time	TIMESTAMP	NOT NULL	DEFAULT now
editor	FOREIGN KEY → Person	NOT NULL	
value	STRING		
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (user, attribute, time)		
Description	This relation tells us that the Person mentioned has the Attribute given.		

personHasPrivilege (Person → Privilege)			
user	FOREIGN KEY → Person	NOT NULL	
privilege	FOREIGN KEY → Privilege	NOT NULL	
ctime	TIMESTAMP	NOT NULL	DEFAULT now
mtime	TIMESTAMP	NOT NULL	DEFAULT now
created_by	FOREIGN KEY → Person		
modified_by	FOREIGN KEY → Person		
target	STRING		
because_of_role	BOOLEAN		
Constraint	PRIMARY KEY (user, privilege, target)		
Description	This relation tells us that the given Person has the given Privilege.		

personHasRole (Person → Role)			
user	FOREIGN KEY → Person	NOT NULL	
role	FOREIGN KEY → Role	NOT NULL	
ctime	TIMESTAMP	NOT NULL	DEFAULT now
mtime	TIMESTAMP	NOT NULL	DEFAULT now
created_by	FOREIGN KEY → Person		
modified_by	FOREIGN KEY → Person		
Constraint	PRIMARY KEY (user, role)		
Description	This relation tells us that the given Person has the given Role.		

personPrefersTimeslot (Person → Timeslot)			
user	FOREIGN KEY → Person	NOT NULL	
timeslot	FOREIGN KEY → Timeslot	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
priority	INTEGER	NOT NULL	0 ≤ priority ≤ 100
Constraint	PRIMARY KEY (user, timeslot)		
Description	This relation describes that the given person wants this Timeslot more than other timeslots.		

personSuccessfullyPassedCourse (Person → Course)			
user	FOREIGN KEY → Person	NOT NULL	
course	FOREIGN KEY → Course	NOT NULL	

ctime	TIMESTAMP	NOT NULL	DEFAULT now
mtime	TIMESTAMP	NOT NULL	DEFAULT now
created_by	FOREIGN KEY → Person		
modified_by	FOREIGN KEY → Person		
grade	STRING		
notes	TEXT		
Constraint	PRIMARY KEY (user, course)		
Description	This relation tells us that a Person has passed a specific Course.		

personTakesPartInElementInstance (Person → CourseElementInstance)			
user	FOREIGN KEY → Person	NOT NULL	
element_instance	FOREIGN KEY → CourseElementInstance	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
timeslot	FOREIGN KEY → Timeslot	NOT NULL	
Constraint	PRIMARY KEY (user, element_instance)		
Constraint	UNIQUE (user, timeslot)		
Description	This relation tells us that the given Person takes part in the mentioned CourseElementInstance.		

roleImpliesAttribute (Role → Attribute)			
role	FOREIGN KEY → Role	NOT NULL	
attribute	FOREIGN KEY → Attribute	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
default	STRING		
required	BOOLEAN	NOT NULL	DEFAULT false
Constraint	PRIMARY KEY (role, attribute)		
Description	This relation keeps track of user-defined Attributes that are implied by a role.		

roleImpliesPrivilege (Role → Privilege)			
role	FOREIGN KEY → Role	NOT NULL	
privilege	FOREIGN KEY → Privilege	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
target	STRING		
Constraint	PRIMARY KEY (role, privilege, target)		
Description	This relation tells us what Privileges are implied by a given Role.		

roomPrefersTimeslot (Room → Timeslot)			
room	FOREIGN KEY → Room	NOT NULL	
timeslot	FOREIGN KEY → Timeslot	NOT NULL	
priority	INTEGER	NOT NULL	0 ≤ priority ≤ 100
timekey	TIMESTAMP	NOT NULL	DEFAULT now
Constraint	PRIMARY KEY (room, timeslot)		
Description	This relation describes that the given Room should be booked at the given Timeslot.		

roomProvidesFeature (Room → Feature)			
room	FOREIGN KEY → Room	NOT NULL	
feature	FOREIGN KEY → Feature	NOT NULL	
timekey	TIMESTAMP	NOT NULL	DEFAULT now
quantity	INTEGER	NOT NULL	≥ 0
Constraint	PRIMARY KEY (room, feature)		
Description	This relation tells us which Feature is provided by this room.		