

## Recent Developments for Data Models – Exercise 3

Monday, June 4, 2012 – 15:30 to 17:00 – Room 36-336

### 1) SQL-invoked Routines Characteristics

SQL-invoked routines fall into three principle classes: procedures, functions and methods. Fill in the following table to summarize the more important differences!

	Procedures	Functions	Methods
Routine invocation			
Associated with specific type?			
Schema of residence			
Routine resolution			
Input parameters?			
Output parameters?			

### 2) SQL-invoked Routines

Reconsider the user-defined structured type hierarchy introduced by the first exercise sheet. Specify SQL-invoked routines for the following purposes.

- a. A user-defined constructor method for the type `PublicationT` that takes the title, the URL, and the publication date as input parameters.
- b. A method called `citeStr` for the type `PublicationT` that returns a character string of the following format: `<title>, <year of publication>`

- c. A method called `citeStr` for the type `ContainedPublT` that overrides the method of its supertype and returns a character string of the following format: `<title>, <first_page> - <last_page>, <year of publication>`
- d. A function called `citeStr` that takes a parameter of the type `PublicationT` and returns a character string of the format described in b.
- e. A function called `citeStr` that takes a parameter of the type `ContainedPublT` and returns a character string of the format described in c.
- f. A function called `publishedBetween` that takes two `Date` parameters and returns those publications that have been published in the indicated time range.
- g. A procedure called `addAuthor` to add an author to a publication. The procedure shall take three input parameters of type `AuthorT`, `PublicationT`, and `Integer` to indicate the position of the author in the list.
- h. A procedure called `publAuthorsKeywords` that takes a publication key as parameter and returns the associated authors and keywords (in two result sets).

### 3) Subject Routine Determination

Function overloading allows creating multiple functions with equal names. After a function invocation, the database manager must decide which of the equally named functions “fits best”. This function is referred to as *subject function*.

Assume that the function `func` is invoked with three parameters of the types `ArticleT`, `INTEGER`, and `CHAR(50)`. Further assume that the current session `PATH` is set to 'A, B' and that execute privileges are granted for all functions.

- a. Consider the following function signatures.
  - 1 `FUNCTION A.foo (ArticleT, SMALLINT, CHAR(50))`
  - 2 `FUNCTION A.func(BookT, INTEGER, VARCHAR(50))`
  - 3 `FUNCTION A.func(PublicationT, REAL, INTEGER)`
  - 4 `FUNCTION C.func(ArticleT, INTEGER, CHAR(50))`
  - 5 `FUNCTION B.func(ContainedPublT, INTEGER)`
  - 6 `FUNCTION A.func(ArticleT, DECIMAL(5, 2), VARCHAR(50))`
  - 7 `FUNCTION A.func(ContainedPublT, INTEGER, CHAR(50))`
  - 8 `FUNCTION B.func(ArticleT, DECIMAL(5,2), DATE)`
  - 9 `FUNCTION B.func(PublicationT, INTEGER, CHAR(50))`
  - 10 `FUNCTION B.func(ArticleT, DOUBLE, CLOB)`

Which steps are performed to determine the subject function? Which functions are eliminated in each of these steps? Which function is eventually determined to be the subject function?

b. Consider the following function signatures.

- 1 FUNCTION A.func(AuthorT, REAL, CHAR(50))
- 2 FUNCTION A.func(ArticleT, DECIMAL(5, 2), CHAR(50))
- 3 FUNCTION B.func(ArticleT, FLOAT, VARCHAR(50))
- 4 FUNCTION A.func(ArticleT, INTEGER, VARCHAR(50))
- 5 FUNCTION A.func(TechReportT, INTEGER)
- 6 FUNCTION B.func(ArticleT, INTEGER, CLOB)
- 7 FUNCTION B.func(ArticleT, SMALLINT, CHAR(50))
- 8 FUNCTION A.func(ContainedPublT, INTEGER, INTEGER)
- 9 FUNCTION B.func(ArticleT, INTEGER, VARCHAR(50))
- 10 FUNCTION A.func(PublicationT, INTEGER, CHAR(50))

Which function is determined to be the subject function?

c. What are the major differences between subject *function* resolution and subject *method* resolution?

#### 4) External SQL-invoked Routines

SQL-invoked routines can either be written in SQL or in any of several general-purpose programming languages, such as C or Java. These routines are referred to as *external* routines. Discuss the advantages and drawbacks of external routines as compared to SQL routines!

#### 5) SQLJ-1

SQLJ-1 allows for the implementation of external routines using the Java Programming language. The Java application may read and modify database data system using JDBC or SQLJ.

Create a java class containing a method that takes a publication key as input parameter and returns the number of keywords assigned to the publication! What steps are required to create an external function based on this method using DB2?

## 6) User-defined Transforms

Transform functions are used to exchange structured type values with host language programs and with external functions and methods. Pairs of such transform functions are called transform groups; the so called `TO SQL` function is invoked when transferring a structured type instance from the host language side of the interface to the SQL side, and the other, called `FROM SQL` function, is used for transferring from the SQL side to the host language side.

Reconsider the `AccessCntT` structured type introduced by the first exercise sheet. Define a pair of user-defined functions to convert instances of this type to a string representation (simply separate the two integer values by a comma) and vice versa! Furthermore, create a transform group using these functions!