

Christian Weber and Jernej Kovse

A Domain-Specific Language for Versioning

Databases and Information Systems Research Group

Dept. of Computer Science

Kaiserslautern University of Technology

P.O. Box 3049

D-67653 Kaiserslautern

E-mail: {c_weber, kovse}@informatik.uni-kl.de

10.1.2004

DS-DDL for versioning

```
create_objecttype_command ::=  
  create_unversioned_objecttype_clause | create_versioned_objecttype_clause |  
  create_workspacetype_clause;
```

```
create_unversioned_objecttype_clause ::=  
  "CREATE UNVERSIONED OBJECTTYPE" objecttypename "("  
  {attribute_clause ","} [attribute_clause]  
  ")";
```

```
create_versioned_objecttype_clause ::=  
  "CREATE VERSIONED OBJECTTYPE" objecttypename "("  
  {attribute_clause ","}  
  "MAXSUCCESSORS" n  
  ")";
```

```
create_workspacetype_clause ::= ;  
  "CREATE VERSIONED OBJECTTYPE" objecttypename "("  
  {attribute_clause ","}  
  {attachment_clause ","}  
  "MAXSUCCESSORS" maxValue_clause  
  ")";
```

```
create_relationship_command ::=  
  "CREATE RELATIONSHIP TYPE" relationshiptypename "("  
  "CONNECTS" objecttypename "(" n "," maxValue_clause ")" [ "FLOATING" ] ","  
  "CONNECTS" objecttypename "(" n "," maxValue_clause ")" [ "FLOATING" ] [ "," ]  
  [ "PROPAGATES" ] proptype { "," proptype } "ON" objecttypename [ "," ]  
  [ "PROPAGATES" ] proptype { "," proptype } "ON" objecttypename  
  ")";
```

```
attribute_clause ::=  
  "ATTRIBUTE" attributename datatype_clause;
```

```
attachment_clause ::=  
  "ATTACHMENT" objecttypename "(" n "," maxValue_clause ")";
```

```
datatype_clause ::=  
  "String" | "Integer" | "Float" | "Boolean" |  
  "Date" | "DateTime" | "BLOB" | "CLOB";
```

```
proptype ::=  
  "attach_detach" | "create_successor" | "freeze" |  
  "checkout_checkin" | "copy" | "new" | "delete";
```

```
objecttypename ::= identifier;
```

```
attributename ::= identifier;
```

```
relationshiptypename ::= identifier;
```

```
maxValue_clause ::= n | "*";
```

```
identifier ::=  
  "letter" { "letter" | "digit" | "_" | "$" | "#" };
```

```
n ::= "digit" { "digit" };
```

DS-DML for versioning

```
new_object_command ::=
  "CREATE NEW OBJECT" ObjectTypeName [ "(" column_list ")"
  "VALUES (" value_list ")" ] [ use_Workspace_clause ];

new_relationship_command ::=
  "CREATE NEW RELATIONSHIP" RelationshipTypeName "WHERE"
  Rolename".GID=" id_value AND Rolename".GID=" id_value;

copy_object_command ::=
  "COPY OBJECT" ( object_select_clause_with_workspace |
  workspace_object_select_clause
  [ "PROPAGATE TO ATTACHED OBJECTS" ] );

createSuccessor_command ::=
  "CREATE SUCCESSOR OF OBJECT" (
  versioned_object_select_clause_with_workspace |
  workspace_object_select_clause
  [ "PROPAGATE TO ATTACHED OBJECTS" ] );

freeze_object_command ::=
  "FREEZE OBJECT" (
  versioned_object_select_clause_with_workspace |
  workspace_object_select_clause
  [ "PROPAGATE TO ATTACHED OBJECTS" ] );

delete_object_command ::=
  "DELETE OBJECT" ( ObjectTypeName "WHERE OID =" id_value
  [ use_Workspace_clause ] | WorkspaceTypeName
  "WHERE OID =" id_value [ "PROPAGATE TO ATTACHED OBJECTS" ] );

delete_relationship_command ::=
  "DELETE RELATIONSHIP RelationshipTypeName WHERE
  Rolename".OID=" id_value AND Rolename".OID=" id_value;

merge_objects_command ::=
  "MERGE OBJECTS" ObjectTypeName "WHERE" ( "OID =" id_value
  "AND PRIM_VID =" id_value "AND SEC_VID =" id_value |
  PRIM_GID =" id_value "AND SEC_GID =" id_value );

update_object_command ::=
  "UPDATE" ObjectTypeName update_clause ( object_select_clause2 |
  "WHERE OID=" id_value [ "AND" condition ] );

select_command ::=
  "SELECT" [ "DISTINCT" ] ( "*" |
  ( displayed_column { "," displayed_column } ) )
  "FROM" from_clause { "," from_clause } [ "WHERE" condition ]
  { group_clause } { set_clause } { order_clause }
  [use_Workspace_clause];

getRoot_command ::=
  "GET ROOT OF" versioned_object_select_clause_with_workspace;

getPredecessor_command ::=
```

```

"GET PREDECESSOR OF"
versioned_object_select_clause_with_workspace;

getSuccessors_command ::=
"GET SUCCESSORS OF"
versioned_object_select_clause_with_workspace;

getAlternatives_command ::=
"GET ALTERNATIVES OF"
versioned_object_select_clause_with_workspace;

getBaseVersion_command ::=
"GET BASEVERSION OF" VersionedObjectName "WHERE" (
"OID =" id_value "AND VID1 =" id_value "AND VID2 =" id_value |
"GID1 =" id_value "AND GID2 =" id_value );

get_cvc_command ::=
"GET CVC FROM" object_select
"ROLENAME" Rolename;

pinning_command ::=
"PIN" versioned_object_select "IN CVC"
object_select "ROLENAME" Rolename;

unpinning_command ::=
"UNPIN" versioned_object_select IN CVC
versioned_object_select "ROLENAME" Rolename;

attach_command ::=
"ATTACH" ObjectTypeName "GID =" id_value
"TO WORKSPACE" WorkspaceTypeName "GID =" id_value;

detach_command ::=
"DETACH" ObjectTypeName "GID =" id_value
"FROM WORKSPACE" WorkspaceTypeName "GID =" id_value;

checkout_command ::=
"CHECKOUT" ObjectTypeName "GID =" id_value
"FROM WORKSPACE" WorkspaceTypeName "GID =" id_value;

checkin_command ::=
"CHECKIN" ObjectTypeName "GID =" id_value
"TO WORKSPACE" WorkspaceTypeName "GID =" id_value;

object_select_clause ::=
unversioned_object_select | versioned_object_select;

object_select_clause_with_workspace ::=
object_select [ use_Workspace_clause ];

unversioned_object_select ::=
UnversionedObjectName "WHERE"
( "OID" | "GID" ) =" id_value;

versioned_object_select ::=
VersionedObjectName "WHERE" ( "GID =" id_value |

```

```

"OID =" id_value "AND VID=" id_value );

versioned_object_select_with_workspace ::=
  versioned_object_select [ use_Workspace_clause ];

workspace_object_select ::=
  WorkspaceTypeName "WHERE" ( "GID =" id_value |
  "OID =" id_value "AND VID=" id_value );

use_Workspace_clause ::=
  "USE WORKSPACE" workspace_object_select;

from_clause ::=
  selected_table { object_navigation_clause };

object_navigation_clause ::=
  "--" Rolename "--> " selected_table;
  select_end_clauses ::= { group_clause } { set_clause } { order_clause };

update_clause ::=
  "SET" column_name "=" expression
  { "," column_name "=" expression };

order_clause ::=
  "ORDER BY" column_name [( "ASC" | "DESC" )]
  { "," column_name [( "ASC" | "DESC" )]};

set_clause ::=
  ( ( "UNION ALL" ) | "INTERSECT" | "MINUS" ) select_command;

group_clause ::=
  "GROUP BY" expression { "," expression } [ "HAVING" condition ];

condition ::= [ "NOT" ] logical_term { "OR" logical_term };

logical_term ::= logical_factor { "AND" logical_factor };

logical_factor ::=
  ( expression comparaison_op exprssion )
  | ( [ "NOT" ] "LIKE" match_string )
  | ( "IS" [ "NOT" ] "null" )
  | ( "(" condition ")" );

displayed_column ::=
  ( ObjectTypeName "." ( "*" | column_name ) ) |
  ( expression [ "AS" alias ] );

selected_table ::= ObjectTypeName [ alias ];

expression ::= [ "+" | "-" ] term { ( "+" | "-" ) term };

comparaison_op ::=
  "=" | "<" | ">" | "<>" | "!=" | "^=" | "<=" | ">=";

match_string ::= "" { "any_character" | "_" | "%" } "";

```

```

column_list ::= column_name { "," column_name };
value_list ::= value { "," value };
ObjectTypeName ::= identifier;
VersionedObjectTypeName ::= identifier;
WorkspaceTypeName ::= identifier;
RelationshipTypeName ::= identifier;
Rolename ::= identifier;
alias ::= identifier;
id_value ::= n { n };
term ::= factor { ( "*" | "/" ) factor };
factor ::=
  n | column_name
  | ( group_function "(" [ "DISTINCT" ] expression ")" );
group_function ::= "AVG" | "COUNT" | "MAX" | "MIN" | "SUM";
value ::= ( [ "+" | "-" ] ( identifier | n ) ) | quoted_string;
n ::= "digit" { "digit" };
identifier ::=
  ( "letter" { "letter" | "digit" | "_" | "$" | "#" } )
  | quoted_string;
quoted_string ::= "" { "any_character" } "";

```