



# Schema Specification in TSQL2

September 16, 1994

A TSQL2 Commentary

The TSQL2 Language Design Committee

Title Schema Specification in TSQL2  
Primary Author(s) Richard Snodgrass, Christian S. Jensen and Fabio Grandi  
Publication History September 1994. A TSQL2 Commentary.

#### TSQL2 Language Design Committee

Richard T. Snodgrass, Chair rts@cs.arizona.edu	University of Arizona Tucson, AZ
Ilsoo Ahn ahn@cbtnmva.att.com	AT&T Bell Laboratories Columbus, OH
Gad Ariav ariavg@ccmail.gsm.uci.edu	Tel Aviv University Tel Aviv, Israel
Don Batory dsb@cs.utexas.edu	University of Texas Austin, TX
James Clifford jcliffor@is-4.stern.nyu.edu	New York University New York, NY
Curtis E. Dyreson curtis@cs.arizona.edu	University of Arizona Tucson, AZ
Ramez Elmasri elmasri@cse.uta.edu	University of Texas Arlington, TX
Fabio Grandi fabio@deis64.cineca.it	Università di Bologna Bologna, Italy
Christian S. Jensen csj@iesd.auc.dk	Aalborg University Aalborg, Denmark
Wolfgang Käfer kaefer%fuzi.uucp@germany.eu.net	Daimler Benz Ulm, Germany
Nick Kline kline@cs.arizona.edu	University of Arizona Tucson, AZ
Krishna Kulkarni kulkarni_krishna@tandem.com	Tandem Computers Cupertino, CA
T. Y. Cliff Leung cleung@vnet.ibm.com	Data Base Technology Institute, IBM San Jose, CA
Nikos Lorentzos eliop@isosun.ariadne-t.gr	Agricultural University of Athens Athens, Greece
John F. Roddick roddick@unisa.edu.au	University of South Australia The Levels, South Australia
Arie Segev segev@csr.lbl.gov	University of California Berkeley, CA
Michael D. Soo soo@cs.arizona.edu	University of Arizona Tucson, AZ
Suryanarayana M. Sripada sripada@ecrc.de	European Computer-Industry Research Centre Munich, Germany

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Informal Definition</b>	<b>1</b>
<b>3</b>	<b>Syntactic Considerations</b>	<b>2</b>
<b>4</b>	<b>Acknowledgements</b>	<b>2</b>
<b>A</b>	<b>Modified Language Syntax</b>	<b>2</b>
A.1	Section 5.2 <token> . . . . .	3
A.2	Section 11.3 <table definition> . . . . .	3
A.3	Section 11.10 <alter table statement> . . . . .	3

## Abstract

This document proposes syntax and informal semantics for extended Create and Alter statements that permit valid-time tables to be defined.

## 1 Introduction

The Create statement should be enhanced to support the definition of valid-time relations. Additionally, the Alter statement should be extended to allow these definitions to be later changed.

In the language extension to be discussed shortly, we adopted the following goals.

1. Extensions should be upward compatible with current SQL2.
2. Extensions should be as minimal as possible.
3. As few reserved words as possible should be introduced.
4. Punctuation should be consistent with the rest of the language.
5. Extensions should be consistent and compatible with user-defined time syntax.
6. The clean lines of the BCDM should be retained.
7. Defaults should be carefully chosen to reflect common usage and to enable a suitable reduction proof (see (1)).

## 2 Informal Definition

We examine several examples of the new Create and Alter statements, to provide a very informal description. As will be seen, this is an extension of the previous syntax.

The `Employee` relation, with `Name`, `Dept`, and `Salary` attributes, would be defined as follows in SQL2.

```
CREATE TABLE Employee (Name CHAR, Dept CHAR, Salary INT)
```

An `ON` clause can be added at the end, e.g.,

```
CREATE TABLE Employee (Name CHAR, Dept CHAR, Salary INT) ON COMMIT PRESERVE ROWS
```

The specification of a valid-time relation is accomplished through an additional clause, placed before the `ON` clause. To specify a valid-time relation,

```
CREATE TABLE Employee (Name CHAR, Dept CHAR, Salary INT) AS VALID
```

The precision and scale of the last component of the AS VALID clause can also be specified, e.g.,

```
CREATE TABLE Employee (Name CHAR, Dept CHAR, Salary INT) AS VALID(11,6)
```

The Alter statement can be used to later modify the schema. The precision can be changed as follows.

```
ALTER TABLE Employee ADD VALID(11,0)
```

Finally, the table can be changed back into a snapshot relation.

```
ALTER TABLE Employee DROP VALID
```

### 3 Syntactic Considerations

We considered and rejected the following alternatives, for the reasons given.

- `CREATE INTERVAL TABLE` is awkward when the precision must be specified, e.g., `CREATE INTERVAL(11,6) TABLE Employee`. Also, this conflicts with the sense in SQL-92 of intervals as *unanchored* durations of time.
- A distinguished attribute, say with name `ValidTime`, is incompatible with the proposed temporal projection syntax.
- As another `<table element>` is inappropriate, because the `<column definition>` and `<table constraint definition>` clauses that make up a `<table element>` both heavily refer to table columns, whereas the valid as clause applies to the table as a whole.

### 4 Acknowledgements

This work was supported in part by NSF grants ISI-8902707 and ISI-9302244, IBM contract #1124 and the AT&T Foundation. Support was also provided in part by Danish Natural Science Research Council grants 11-1089-1 SE and 11-0061-1 SE.

Curtis Dyreson, Nick Kline and Michael Soo provided helpful comments on previous drafts.

### A Modified Language Syntax

The organization of this section follows that of the SQL2 document. The syntax is listed under corresponding section numbers in the SQL2 document. All new or modified syntax rules are marked with a bullet (“•”) on the left side of the production.

Where appropriate, we provide disambiguating rules to describe additional syntactic and semantic restrictions. We assume that the reader is familiar with the SQL2 proposal, and that a copy of the proposal is available for reference.

### A.1 Section 5.2 <token>

One reserved word was added.

<reserved word> ::=

- | VALID

### A.2 Section 11.3 <table definition>

The production for the non-terminal <table definition> is augmented with an additional, optional clause.

<table definition> ::=

- CREATE [ { GLOBAL | LOCAL } TEMPORARY ] TABLE <table name>  
    <table elements>  
    [ <temporal definition> ]  
    [ ON COMMIT { DELETE | PRESERVE } ] ROWS ]

One production is added.

<temporal definition> ::=

- AS VALID [ <timestamp precision> ]

Additional general rules:

1. If <valid definition> is not specified, the table is a snapshot table.
2. If <valid definition> is specified, the table is a valid-time table.
3. If AS VALID is specified, then the tuples are timestamped with temporal elements that are sets of non-contiguous periods of time. The precision and scale of the timestamps can be specified.

### A.3 Section 11.10 <alter table statement>

The <alter table statement> is augmented with the following alternatives.

<alter table action> ::=

- <add valid definition>
- <drop valid definition>
- <replace valid definition>

The following productions are added.

<add valid definition> ::=

- ADD VALID [ <timestamp precision> ]

Additional syntax rules:

1. Let T be the table identified in the containing <alter table statement>.
2. T shall be a snapshot table.

<drop valid definition> ::=

- DROP VALID

Additional syntax rules:

1. Let T be the table identified in the containing <alter table statement>.
2. T shall be a valid-time table.

Additional general rules:

1. T is converted to a snapshot table, with contents

```
SELECT SNAPSHOT * FROM T WHERE T OVERLAPS CURRENT_TIMESTAMP
```

<replace valid definition> ::=

- REPLACE VALID [ <time precision> ]