Specifier Syntax

Specifiers are objects which specify schema or data elements.

In general a specifier is an object with a `type` property indicating the type of the specifier. Some specifiers provide a shorthand version by substituting a different JSON data type.

Common

There are several common specifiers used by multiple components.

All

Specifies all elements in the current context (for example, as a records specifier all would include all records in the selected database).

Property	Value
`type`	`"all"`

Example

```
{ "type": "all" }
```

ID

Specifies an element by numeric ID. The value is provided directly as a JSON number.

Example (JSON number)

```
123
```

Name

Specifies an element by name. The value is provided directly as a JSON string.

Example (JSON string)

```
"foo"
```

Where

Specifies element(s) meeting a condition provided by an expression.

The source against which the expression is used depends on the context, but in general can be represented as

```
SELECT [elements] FROM [source] WHERE [expression]
```

where source is the table containing the element.

Property	Value
`type`	`"where"`
`where`	expression

Array

Specifies elements using an array of singular specifiers. The value is provided directly as a JSON array.

The types of the individual specifiers depend on the element, but in general unless otherwise noted all singular specifier types for the element may be used. Specifier types may also be intermingled.

Example (JSON array)

```
[ 123, "foo", {"type": "where", "where", "..."} ]
```

Schema Elements

Groups

Specifies one or more groups. Group specifiers are also valid groups specifiers.

- All
- Array

Group

Specifies a single group.

- ID
- Name

Databases

Specifies one or more databases. Database specifiers are also valid databases specifiers.

- All
- Array

Database

Specifies a single database.

- ID
- Name

Fields

Specifies one or more fields. Field specifiers are also valid fields specifiers.

- All
- Array

Field

Specifies a single field.

- ID
- Name

Walls

Specifies one or more walls. Wall specifiers are also valid walls specifiers.

- `Array`

Wall

Specifies a single wall.

Group Wall

Specifies the wall of single group.

Property	Value
`type`	`"group"`
`group`	group specifier

Example

```
{
 "type"  : "group",
 "group" : "foo"
}
```

Database Wall

Specifies the wall of single database.

Property	Value
`type`	`"database"`
`database`	database specifier

Example

```
{
 "type"     : "database",
 "database" : "foo"
}
```

Record Wall

Specifies the wall of single record.

Property	Value
`type`	`"record"`
`database`	database specifier
`record`	record specifier

Example

```
{
 "type"     : "database",
 "database" : "foo",
 "record"   : 123
```

```
}
```

User Wall

Specifies the wall of single user.

Property	Value
`type`	`"user"`
`user`	user specifier

Example

```
{
 "type" : "user",
 "user" : "foo"
}
```

Data Elements

Records

Specifies a set of records in a single database. Record specifiers are also valid records specifiers.

- All
- Array
- Where

Record

Specifies a single record in a database.

- `ID`

Key

Specifies a single record by a set of key value(s).

{ "type" : "key", "key" : <[[XINA API :: Data Syntax#Fields|fields]]> }

Each key must specify a non-null value for each key field of the database.

Posts

Specifies a set of posts. Post specifiers are also valid posts specifiers.

- `All`
- `Array`
- `Where`

Post

Specifies a single post.

- `ID`

Administrative

Users

Specifies a set of users. User specifiers are also valid users specifiers.

- `All`
- `Array`
- `Where`

User

Specifies a single user. Note that `name` in this case refers to the username, not the user's full name.

- `ID`
- `Name`

Group Privileges

Specifies a set of group privileges.

- `All`
- `Array`

Group Privilege

Specifies a single group privilege as a JSON string. The valid group privileges are:

- "select"
- "post"
- "reply"
- "alter"
- "grant"

Database Privileges

Specifies a set of database privileges.

- All
- Array

Database Privilege

Specifies a single database privilege as a JSON string. The valid database privileges are:

- "select"
- "post"
- "reply"
- "update"
- "insert"
- "trash"
- "delete"
- "lock"
- "alter"
- "grant"

Revision #11
Created 9 June 2022 16:20:24 by Nick Dobson
Updated 15 January 2026 20:43:45 by Bradley Tse