#### `$where` Must Be Top-Level

**Description** `$where` expressions can now only be at top level and cannot be nested within another expression, such as `$elemMatch`.

**Solution** Update existing queries that nest `$where`.

**`$exists` and `notablescan`** If the MongoDB server has disabled collection scans, i.e. `notablescan`, then `$exists` queries that have no *indexed solution* will error.

#### `MinKey` and `MaxKey` Queries

**Description** Equality match for either `MinKey` or `MaxKey` no longer match documents missing the field.

#### Nested Array Queries with $elemMatch

**Description** The `$elemMatch` query operator no longer traverses recursively into nested arrays.

For example, if a collection `test` contains the following document:

```
{ "_id": 1, "a" : [ [ 1, 2, 5 ] ] }
```

In 2.6, the following `$elemMatch` query does *not* match the document:

```
db.test.find( { a: { $elemMatch: { $gt: 1, $lt: 5 } } } )
```

**Solution** Update existing queries that rely upon the old behavior.

**Text Search Compatibility** MongoDB does not support the use of the `$text` query operator in mixed sharded cluster deployments that contain both version 2.4 and version 2.6 shards. See *Upgrade MongoDB to 2.6* (page 866) for upgrade instructions.

#### Replica Set/Sharded Cluster Validation

#### Shard Name Checks on Metadata Refresh

**Description** For sharded clusters, MongoDB 2.6 disallows a shard from refreshing the metadata if the shard name has not been explicitly set.

For mixed sharded cluster deployments that contain both version 2.4 and version 2.6 shards, this change can cause errors when migrating chunks **from** version 2.4 shards **to** version 2.6 shards if the shard name is unknown to the version 2.6 shards. MongoDB does not support migrations in mixed sharded cluster deployments.

**Solution** Upgrade all components of the cluster to 2.6. See *Upgrade MongoDB to 2.6* (page 866).

#### Replica Set Vote Configuration Validation

**Description** MongoDB now deprecates giving any *replica set* member more than a single vote. During configuration, `local.system.replset.members[n].votes` should only have a value of 1 for voting members and 0 for non-voting members. MongoDB treats values other than 1 or 0 as a value of 1 and produces a warning message.

**Solution** Update `local.system.replset.members[n].votes` with values other than 1 or 0 to 1 or 0 as appropriate.