

## TYPO3 Core - Feature #84115

### Missing notInSet equivalent of inSet comparison

2018-03-02 12:25 - Andreas Allacher

<b>Status:</b>	Closed	<b>Start date:</b>	2018-03-02
<b>Priority:</b>	Should have	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	100%
<b>Category:</b>	Database API (Doctrine DBAL)	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>Complexity:</b>	
<b>PHP Version:</b>	7.0	<b>Sprint Focus:</b>	Needs Decision
<b>Tags:</b>			
<b>Description</b>			
<p>Sometimes one needs to check if a value is NOT in a set, so the opposite of inSet. The DBAL expressionbuilder provides "like" and "notLike" etc. However, for inSet that is not available. It should probably be enough to just do something like</p> <pre>'NOT (' . inSet(...) . ')'</pre> <p>to achieve this</p>			

#### Associated revisions

##### Revision cf58a265 - 2021-08-04 10:48 - Benni Mack

[FEATURE] Add "notInSet" to Doctrine DBAL ExpressionBuilder

The "inSet" method is very useful for finding CSV values in fields, however there was never a "notInSet" method which is now available for TYPO3 Core.

Resolves: #84115

Releases: master

Change-Id: led84fa51cb37391c9b814a2cddd5dbeb212fdd44

Reviewed-on: <https://review.typo3.org/c/Packages/TYPO3.CMS/+70152>

Tested-by: core-ci <[typo3@b13.com](mailto:typo3@b13.com)>

Tested-by: Georg Ringer <[georg.ringer@gmail.com](mailto:georg.ringer@gmail.com)>

Tested-by: Oliver Bartsch <[bo@cedev.de](mailto:bo@cedev.de)>

Tested-by: Benni Mack <[benni@typo3.org](mailto:benni@typo3.org)>

Reviewed-by: Georg Ringer <[georg.ringer@gmail.com](mailto:georg.ringer@gmail.com)>

Reviewed-by: Oliver Bartsch <[bo@cedev.de](mailto:bo@cedev.de)>

Reviewed-by: Benni Mack <[benni@typo3.org](mailto:benni@typo3.org)>

#### History

##### #1 - 2018-03-02 12:28 - Andreas Fernandez

- Status changed from New to Rejected

Hi, thank you for your request.

The QueryBuilder indeed support this by using \$queryBuilder->expr()->notIn(). Thus, I will close this ticket.

Kind regards,  
Andreas

##### #2 - 2018-03-03 09:07 - Andreas Allacher

Hi Andreas,

how should \$queryBuilder->expr()->notIn()  
be the same as a negate of \$queryBuilder->expr()->inSet() ?

```

return $this->comparison(
    $this->connection->quoteIdentifier($fieldName),
    'NOT IN',
    '(' . implode(', ', (array)$value) . ')'
);

```

This is notIn whereas inSet looks like this:

```

public function inSet(string $fieldName, string $value, bool $isColumn = false): string
{
    if ($value === '') {
        throw new \InvalidArgumentException(
            'ExpressionBuilder::inSet() can not be used with an empty string value.',
            1459696089
        );
    }
    if (strpos($value, ',') !== false) {
        throw new \InvalidArgumentException(
            'ExpressionBuilder::inSet() can not be used with values that contain a comma (",").',
            1459696090
        );
    }
    switch ($this->connection->getDatabasePlatform()->getName()) {
        case 'postgresql':
        case 'pdo_postgresql':
            return $this->comparison(
                $isColumn ? $value . '::text' : $this->literal($this->unquoteLiteral((string)$value)),
                self::EQ,
                sprintf(
                    'ANY(string_to_array(%s, %s))',
                    $this->connection->quoteIdentifier($fieldName) . '::text',
                    $this->literal(',')
                )
            );
            break;
        case 'oci8':
        case 'pdo_oracle':
            throw new \RuntimeException(
                'FIND_IN_SET support for database platform "Oracle" not yet implemented.',
                1459696680
            );
            break;
        case 'sqlsrv':
        case 'pdo_sqlsrv':
        case 'mssql':
            // See unit and functional tests for details
            if ($isColumn) {
                $expression = $this->orX(
                    $this->eq($fieldName, $value),
                    $this->like($fieldName, $value . ' + \'%,%\''),
                    $this->like($fieldName, '\'' . $value),
                    $this->like($fieldName, '\'' . $value . ' + \'%,%\'')
                );
            } else {
                $likeEscapedValue = str_replace(
                    ['%', '%'],
                    ['[%]', '[%]'],
                    $this->unquoteLiteral($value)
                );
                $expression = $this->orX(
                    $this->eq($fieldName, $this->literal($this->unquoteLiteral((string)$value))),
                    $this->like($fieldName, $this->literal($likeEscapedValue . ',%')),
                    $this->like($fieldName, $this->literal('%' . $likeEscapedValue)),
                    $this->like($fieldName, $this->literal('%' . $likeEscapedValue . ',%'))
                );
            }
            return (string)$expression;
        case 'sqlite':
        case 'sqlite3':
        case 'pdo_sqlite':
            if (strpos($value, ':') === 0 || $value === '?') {
                throw new \InvalidArgumentException(
                    'ExpressionBuilder::inSet() for SQLite can not be used with placeholder arguments.',
                    1476029421
                );
            }
    }
}

```

```

    );
}
return $this->comparison(
    implode('||', [
        $this->literal(',','),
        $this->connection->quoteIdentifier($fieldName),
        $this->literal(',','),
    ]),
    'LIKE',
    $this->literal(
        '%,' . $this->unquoteLiteral($value) . ',%'
    )
);
break;
default:
return sprintf(
    'FIND_IN_SET(%s, %s)',
    $value,
    $this->connection->quoteIdentifier($fieldName)
);
}
}

```

I am quite sure that nIn shouldn't produce the same result.  
Even if: There should still be a method notInSet that then just calls notIn.

### #3 - 2018-03-05 08:11 - Andreas Fernandez

- Status changed from Rejected to New

Sorry, I misunderstood the purpose. I've re-opened the ticket again.

### #4 - 2018-03-05 08:26 - Andreas Allacher

An example for this method would be to check if a backend user is not in a certain usergroup - where usergroups are stored as a comma separated list.

### #5 - 2018-03-05 11:40 - Wouter Wolters

- Target version deleted (8.7.11)

### #6 - 2018-03-05 13:35 - Oliver Hader

- Subject changed from DBAL: missing notInSet equivalent of inSet comparison to Missing notInSet equivalent of inSet comparison

### #7 - 2020-03-17 09:34 - Susanne Moog

- Sprint Focus set to PRC

### #8 - 2020-03-24 17:52 - Benni Mack

- Sprint Focus changed from PRC to Needs Decision

### #9 - 2021-07-29 15:55 - Gerrit Code Review

- Status changed from New to Under Review

Patch set 1 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.  
It is available at <https://review.typo3.org/c/Packages/TYPO3.CMS/+70152>

### #10 - 2021-07-30 09:57 - Gerrit Code Review

Patch set 2 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.  
It is available at <https://review.typo3.org/c/Packages/TYPO3.CMS/+70152>

### #11 - 2021-08-02 10:04 - Gerrit Code Review

Patch set 3 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.  
It is available at <https://review.typo3.org/c/Packages/TYPO3.CMS/+70152>

### #12 - 2021-08-02 11:23 - Gerrit Code Review

Patch set 4 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.  
It is available at <https://review.typo3.org/c/Packages/TYPO3.CMS/+70152>

**#13 - 2021-08-04 10:56 - Benni Mack**

- Status changed from *Under Review* to *Resolved*

- % Done changed from 0 to 100

Applied in changeset [cf58a265c46b9c6edf57e836a3647ffc786ca6ee](#).

**#14 - 2021-09-07 07:37 - Benni Mack**

- Status changed from *Resolved* to *Closed*