



Open Source Used In eXpertOnDemand 1.3

Cisco Systems, Inc.

www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Text Part Number: 78EE117C99-200160341

This document contains licenses and notices for open source software used in this product. With respect to the free/open source software listed in this document, if you have any questions or wish to receive a copy of any source code to which you may be entitled under the applicable free/open source license(s) (such as the GNU Lesser/General Public License), please contact us at external-opensource-requests@cisco.com.

In your requests please include the following reference number 78EE117C99-200160341

Contents

[1.1 Logback 1.1.7](#)

[1.1.1 Available under license](#)

[1.2 realm-java 5.11.0](#)

[1.2.1 Available under license](#)

[1.3 retrofit 1.5.0](#)

[1.3.1 Available under license](#)

1.1 Logback 1.1.7

1.1.1 Available under license :

Logback LICENSE

Logback: the reliable, generic, fast and flexible logging framework.
Copyright (C) 1999-2015, QOS.ch. All rights reserved.

This program and the accompanying materials are dual-licensed under either the terms of the Eclipse Public License v1.0 as published by the Eclipse Foundation

or (per the licensee's choosing)

under the terms of the GNU Lesser General Public License version 2.1 as published by the Free Software Foundation.

1.2 realm-java 5.11.0

1.2.1 Available under license :

TABLE OF CONTENTS

1. Apache License version 2.0
2. Realm Components
3. Export Compliance

1. -----

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

2. -----

REALM COMPONENTS

This software contains components with separate copyright and license terms. Your use of these components is subject to the terms and conditions of the following licenses.

For the Realm Platform Extensions component

Realm Platform Extensions License

Copyright (c) 2011-2017 Realm Inc All rights reserved

Redistribution and use in binary form, with or without modification, is permitted provided that the following conditions are met:

1. You agree not to attempt to decompile, disassemble, reverse engineer or otherwise discover the source code from which the binary code was derived. You may, however, access and obtain a separate license for most of the source code from which this Software was created, at <http://realm.io/pricing/>.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

3. -----

EXPORT COMPLIANCE

You understand that the Software may contain cryptographic functions that may be subject to export restrictions, and you represent and warrant that you are not

(i) located in a jurisdiction that is subject to United States economic sanctions (Prohibited Jurisdiction), including Cuba, Iran, North Korea, Sudan, Syria or the Crimea region, (ii) a person listed on any U.S. government blacklist (to include the List of Specially Designated Nationals and Blocked Persons or the Consolidated Sanctions List administered by the U.S. Department of the Treasury's Office of Foreign Assets Control, or the Denied Persons List or Entity List administered by the U.S. Department of Commerce)

(Sanctioned Person), or (iii) controlled or 50% or more owned by a Sanctioned Person.

You agree to comply with all export, re-export and import restrictions and regulations of the U.S. Department of Commerce or other agency or authority of the United States or other applicable countries. You also agree not to transfer, or authorize the transfer of, directly or indirectly, of the Software to any Prohibited Jurisdiction, or otherwise in violation of any such restrictions or regulations.

/*

* Copyright 2017 Realm Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

package io.realm.permissions;

import java.util.Date;

import edu.umd.cs.findbugs.annotations.SuppressFBWarnings;

import io.realm.PermissionManager;

import io.realm.RealmObject;

import io.realm.SyncUser;

import io.realm.annotations.Required;

/**

* This class represents a given set of permissions for one user on one Realm.

* <p>

* Permissions can be changed by users with administrative rights using the { @link PermissionManager }.

*

* @see SyncUser#getPermissionManager()

*/

public class Permission extends RealmObject {

 @Required

 private String userId;

 @Required


```

private String path;
private boolean mayRead;
private boolean mayWrite;
private boolean mayManage;
@Required
private Date updatedAt;

/**
 * Required by Realm. Do not use.
 */
public Permission() {
    // Required by Realm
}

/**
 * Returns the { @link SyncUser#getIdentity() } of the user effected by this permission.
 * <p>
 *
 * @return the user effected by this permission.
 */
public String getUserId() {
    return userId;
}

/**
 * Returns the path to the Realm on the server effected by this permission. This is not the full URL.
 *
 * @return the path to the Realm this permission object refers to.
 */
public String getPath() {
    return path;
}

/**
 * Checks whether or not the user defined by this permission is allowed to read the Realm defined by
 * { @link #getPath() }.
 *
 * @return { @code true } if this permission grant read permissions to the Realm, { @code false } if not.
 */
public boolean mayRead() {
    return mayRead;
}

/**
 * Checks whether or not the user defined by this permission is allowed to write to the Realm defined by
 * { @link #getPath() }.
 *
 * @return { @code true } if this permission grant write permissions to the Realm, { @code false } if not.

```



```

*/
public boolean mayWrite() {
    return mayWrite;
}

/**
 * Checks whether or not the user defined by this permission is allowed to manage access to the Realm defined
 * by {@link #getPath()}. Having this permission enable those users to add or remove permissions from
 * other users, including the one who granted it.
 *
 * @return {@code true} if this permission grant administrative rights to the Realm, {@code false} if not.
 */
public boolean mayManage() {
    return mayManage;
}

/**
 * Returns the timestamp for when this permission object was last updated.
 *
 * @return the timestamp for when this permission was last updated.
 */
@SuppressFBWarnings({"EI_EXPOSE_REP"})
public Date getUpdatedAt() {
    return updatedAt;
}

@Override
public String toString() {
    return "Permission{" +
        "userId=" + userId + "\" +
        ", path=" + path + "\" +
        ", mayRead=" + mayRead +
        ", mayWrite=" + mayWrite +
        ", mayManage=" + mayManage +
        ", updatedAt=" + updatedAt +
        "'";
}
}

/*
 * Copyright 2018 Realm Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software

```



```

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

package io.realm.sync.permissions;

import io.realm.RealmObject;
import io.realm.annotations.RealmClass;
import io.realm.internal.annotations.ObjectServer;

/**
 * This class encapsulates the privileges granted a given {@link Role}. These privileges can be
 * applied to either the entire Realm, Classes or individual objects.
 * <p>
 * If no privileges are defined for an individual object, the values {@link ClassPermissions}
 * will be inherited, if no values are defined there, the ones from {@link RealmPermissions} will
 * be used. If no values can be found there, no privileges are granted.
 * <p>
 * Not all privileges are meaningful all levels, e.g. `canCreate` is only meaningful when applied to
 * classes, but it can still be defined at the Realm level. In that case all class permission objects
 * will inherit the value unless they specifically override it. See the individual privileges for the
 * details.
 * <p>
 * When added to either {@link RealmPermissions}, {@link ClassPermissions} or a {@link RealmObject},
 * only one Permission object can exist for that role. If multiple objects are added the behavior
 * is undefined and the Object Server might modify or delete both objects.
 *
 * @see <a href="FIX">Object Level Permissions</a> for an detailed description of the Realm Object
 * Server permission system.
 */
@ObjectServer
@RealmClass(name = "__Permission")
public class Permission extends RealmObject {

    /**
     * Creates a {@link Permission} object in a fluid manner.
     */

    public static class Builder {
        private Role role;
        private boolean canRead = false;
        private boolean canUpdate = false;
        private boolean canDelete = false;
        private boolean canSetPermissions = false;
        private boolean canQuery = false;
        private boolean canCreate = false;
        private boolean canModifySchema = false;

```



```

/**
 * Creates the builder. The default state is that no privileges are enabled.
 *
 * @param role { @link Role } for which these privileges apply.
 */
public Builder(Role role) {
    this.role = role;
}

/**
 * Enables all privileges.
 */
public Builder allPrivileges() {
    canRead = true;
    canUpdate = true;
    canDelete = true;
    canSetPermissions = true;
    canQuery = true;
    canCreate = true;
    canModifySchema = true;
    return this;
}

/**
 * Disables all privileges.
 */
public Builder noPrivileges() {
    canRead = false;
    canUpdate = false;
    canDelete = false;
    canSetPermissions = false;
    canQuery = false;
    canCreate = false;
    canModifySchema = false;
    return this;
}

/**
 * Defines if this role can read from given resource or not.
 *
 * <ol>
 *   <li>
 *     <b>Realm:</b>
 *     The role is allowed to read all objects from the Realm. If { @code false }, the
 *     Realm will appear completely empty to the role, effectively making it inaccessible.
 *   </li>
 *   <li>
 *     <b>Class:</b>

```



```

*   The role is allowed to read the objects of this type and all referenced objects,
*   even if those objects themselves have set this to { @code false}.
*   If { @code false}, the role cannot see any object of this type and all queries
*   against the type will return no results.
* </li>
* <li>
*   <b>Object:</b>
*   Determines if a role is allowed to see the individual object or not.
* </li>
* </ol>
*
* @param canRead { @code true} if the role is allowed to read this resource, { @code false} if not.
*/
public Builder canRead(boolean canRead) {
    this.canRead = canRead;
    return this;
}

/**
* Defines if this role can update the given resource or not.
*
* <ol>
*   <li>
*       <b>Realm:</b>
*       If { @code true}, the role is allowed update properties on all objects in the Realm.
*       This does not include updating permissions nor creating or deleting objects.
*   </li>
*   <li>
*       <b>Class:</b>
*       If { @code true}, the role is allowed update properties on all objects of this type in
*       the Realm. This does not include updating permissions nor creating or deleting objects.
*   </li>
*   <li>
*       <b>Object:</b>
*       If { @code true}, the role is allowed to update properties on the object. This
*       does not cover updating permissions or deleting the object.
*   </li>
* </ol>
*
* @param canUpdate { @code true} if the role is allowed to update this resource, { @code false} if not.
*/
public Builder canUpdate(boolean canUpdate) {
    this.canUpdate = canUpdate;
    return this;
}

/**
* Defines if this role can delete the given resource or not.

```



```

*
* <ol>
*   <li>
*     <b>Realm:</b>
*     Not applicable.
*   </li>
*   <li>
*     <b>Class:</b>
*     Not applicable.
*   </li>
*   <li>
*     <b>Object:</b>
*     If { @code true }, the role is allowed to delete the object.
*   </li>
* </ol>
*
* @param canDelete { @code true } if the role is allowed to delete this resource, { @code false } if not.
*/
public Builder canDelete(boolean canDelete) {
    this.canDelete = canDelete;
    return this;
}

/**
* Defines if this role is allowed to change permissions on the given resource.
* Permissions can only be granted at the same permission level or below. E.g. if set on
* a Class, it is not possible to change Realm level permissions, but does allow the role to
* change object level permissions for objects of that type.
*
* <ol>
*   <li>
*     <b>Realm:</b>
*     The role is allowed to modify the { @link RealmPermissions } object.
*   </li>
*   <li>
*     <b>Class:</b>
*     The role is allowed the change the { @link ClassPermissions } object.
*   </li>
*   <li>
*     <b>Object:</b>
*     The role is allowed to change the permissions on this object.
*   </li>
* </ol>
*
* @param canSetPermissions { @code true } if the role is allowed to change the permissions for this resource.
*/
public Builder canSetPermissions(boolean canSetPermissions) {
    this.canSetPermissions = canSetPermissions;

```



```

    return this;
}

/**
 * Defines if this role is allowed to query the resource or not.
 * <p>
 * Note, that local queries are always possible, but the query result will just be empty.
 *
 * <ol>
 *   <li>
 *     <b>Realm:</b>
 *     Not applicable.
 *   </li>
 *   <li>
 *     <b>Class:</b>
 *     The role is allowed to query objects of this type.
 *   </li>
 *   <li>
 *     <b>Object:</b>
 *     Not applicable.
 *   </li>
 * </ol>
 *
 * @param canQuery { @code true } if the role is allowed to query objects of this type.
 */
public Builder canQuery(boolean canQuery) {
    this.canQuery = canQuery;
    return this;
}

/**
 * Defines if this role is allowed to create objects of this type.
 *
 * <ol>
 *   <li>
 *     <b>Realm:</b>
 *     Not applicable.
 *   </li>
 *   <li>
 *     <b>Class:</b>
 *     If { @code true }, the role is allowed to create objects of this type.
 *   </li>
 *   <li>
 *     <b>Object:</b>
 *     Not applicable.
 *   </li>
 * </ol>

```



```

*
* @param canCreate { @code true } if the role is allowed to create objects of this type.
*/
public Builder canCreate(boolean canCreate) {
    this.canCreate = canCreate;
    return this;
}

/**
 * Defines if this role is allowed to modify the schema of this resource.
 *
 * <ol>
 *   <li>
 *     <b>Realm:</b>
 *     If { @code true } the role is allowed to create classes in the Realm.
 *   </li>
 *   <li>
 *     <b>Class:</b>
 *     If { @code true }, the role is allowed to add properties to the specified class.
 *   </li>
 *   <li>
 *     <b>Object:</b>
 *     Not applicable.
 *   </li>
 * </ol>
 *
 * @param canModifySchema { @code true } if the role is allowed to modify the schema of this resource.
*/
public Builder canModifySchema(boolean canModifySchema) {
    this.canModifySchema = canModifySchema;
    return this;
}

/**
 * Creates the unmanaged { @link Permission } object.
*/
public Permission build() {
    return new Permission(
        role,
        canRead,
        canUpdate,
        canDelete,
        canSetPermissions,
        canQuery,
        canCreate,
        canModifySchema
    );
}

```



```

    }

    private Role role;
    private boolean canRead;
    private boolean canUpdate;
    private boolean canDelete;
    private boolean canSetPermissions;
    private boolean canQuery;
    private boolean canCreate;
    private boolean canModifySchema;

    public Permission() {
        // Required by Realm
    }

    /**
     * Creates a set of privileges for the given role.
     */
    public Permission(Role role) {
        this.role = role;
    }

    /**
     * Creates a set of privileges for the given role.
     */
    private Permission(Role role, boolean canRead, boolean canUpdate, boolean canDelete, boolean
canSetPermissions, boolean canQuery, boolean canCreate, boolean canModifySchema) {
        this.role = role;
        this.canRead = canRead;
        this.canUpdate = canUpdate;
        this.canDelete = canDelete;
        this.canSetPermissions = canSetPermissions;
        this.canQuery = canQuery;
        this.canCreate = canCreate;
        this.canModifySchema = canModifySchema;
    }

    /**
     * Returns the role these privileges apply to.
     *
     * @return the role these privileges apply to.
     */
    public Role getRole() {
        return role;
    }

    /**
     * Returns { @code true } if the role is allowed to read the resource, { @code false } if not.

```



```

*/
public boolean canRead() {
    return canRead;
}

/**
 * Defines if this role can read from given resource or not.
 *
 * <ol>
 *   <li>
 *     <b>Realm:</b>
 *     The role is allowed to read all objects from the Realm. If { @code false }, the
 *     Realm will appear completely empty to the role, effectively making it inaccessible.
 *   </li>
 *   <li>
 *     <b>Class:</b>
 *     The role is allowed to read the objects of this type and all referenced objects,
 *     even if those objects themselves have set this to { @code false }.
 *     If { @code false }, the role cannot see any object of this type and all queries
 *     against the type will return no results.
 *   </li>
 *   <li>
 *     <b>Object:</b>
 *     Determines if a role is allowed to see the individual object or not.
 *   </li>
 * </ol>
 *
 * @param canRead { @code true } if the role is allowed to read this resource, { @code false } if not.
 */
public void setCanRead(boolean canRead) {
    this.canRead = canRead;
}

/**
 * Returns { @code true } if the role is allowed to update the resource, { @code false } if not.
 */
public boolean canUpdate() {
    return canUpdate;
}

/**
 * Defines if this role can update the given resource or not.
 *
 * <ol>
 *   <li>
 *     <b>Realm:</b>
 *     If { @code true }, the role is allowed update properties on all objects in the Realm.
 *     This does not include updating permissions nor creating or deleting objects.

```



```

* </li>
* <li>
*   <b>Class:</b>
*   If { @code true }, the role is allowed update properties on all objects of this type in
*   the Realm. This does not include updating permissions nor creating or deleting objects.
* </li>
* <li>
*   <b>Object:</b>
*   If { @code true }, the role is allowed to update properties on the object. This
*   does not cover updating permissions or deleting the object.
* </li>
* </ol>
*
* @param canUpdate { @code true } if the role is allowed to update this resource, { @code false } if not.
*/
public void setCanUpdate(boolean canUpdate) {
    this.canUpdate = canUpdate;
}

/**
* Returns { @code true } if the role is allowed to delete the object , { @code false } if not.
*/
public boolean canDelete() {
    return canDelete;
}

/**
* Defines if this role can delete the given resource or not.
*
* <ol>
*   <li>
*     <b>Realm:</b>
*     Not applicable.
*   </li>
*   <li>
*     <b>Class:</b>
*     Not applicable.
*   </li>
*   <li>
*     <b>Object:</b>
*     If { @code true }, the role is allowed to delete the object.
*   </li>
* </ol>
*
* @param canDelete { @code true } if the role is allowed to delete this resource, { @code false } if not.
*/
public void setCanDelete(boolean canDelete) {
    this.canDelete = canDelete;
}

```



```

}

/**
 * Returns { @code true } if this this role is allowed to change permissions on the given resource.
 */
public boolean canSetPermissions() {
    return canSetPermissions;
}

/**
 * Defines if this role is allowed to change permissions on the given resource.
 * Permissions can only be granted at the same permission level or below. E.g. if set on
 * a Class, it is not possible to change Realm level permissions, but does allow the role to
 * change object level permissions for objects of that type.
 *
 * <ol>
 *   <li>
 *     <b>Realm:</b>
 *     The role is allowed to modify the { @link RealmPermissions } object.
 *   </li>
 *   <li>
 *     <b>Class:</b>
 *     The role is allowed the change the { @link ClassPermissions } object.
 *   </li>
 *   <li>
 *     <b>Object:</b>
 *     The role is allowed to change the permissions on this object.
 *   </li>
 * </ol>
 *
 * @param canSetPermissions { @code true } if the role is allowed to change the permissions for this resource.
 */
public void setCanSetPermissions(boolean canSetPermissions) {
    this.canSetPermissions = canSetPermissions;
}

/**
 * Returns { @code true } if the role is allowed to query the resource, { @code false } if not.
 */
public boolean canQuery() {
    return canQuery;
}

/**
 * Defines if this role is allowed to query the resource or not.
 * <p>
 * Note, that local queries are always possible, but the query result will just be empty.
 *

```



```

* <ol>
*   <li>
*     <b>Realm:</b>
*     Not applicable.
*   </li>
*   <li>
*     <b>Class:</b>
*     The role is allowed to query objects of this type.
*   </li>
*   <li>
*     <b>Object:</b>
*     Not applicable.
*   </li>
* </ol>
*
* @param canQuery { @code true } if the role is allowed to query objects of this type.
*/
public void setCanQuery(boolean canQuery) {
    this.canQuery = canQuery;
}

/**
* Returns { @code true } if the role is allowed to create objects, { @code false } if not.
*/
public boolean canCreate() {
    return canCreate;
}

/**
* Defines if this role is allowed to create objects of this type.
*
* <ol>
*   <li>
*     <b>Realm:</b>
*     Not applicable.
*   </li>
*   <li>
*     <b>Class:</b>
*     If { @code true }, the role is allowed to create objects of this type.
*   </li>
*   <li>
*     <b>Object:</b>
*     Not applicable.
*   </li>
* </ol>
*
* @param canCreate { @code true } if the role is allowed to create objects of this type.
*/

```



```

public void setCanCreate(boolean canCreate) {
    this.canCreate = canCreate;
}

/**
 * Returns { @code true } if the role is allowed to modify the schema of the resource,
 * { @code false } if not.
 */
public boolean canModifySchema() {
    return canModifySchema;
}

/**
 * Defines if this role is allowed to modify the schema of this resource.
 *
 * <ol>
 *   <li>
 *     <b>Realm:</b>
 *     If { @code true } the role is allowed to create classes in the Realm.
 *   </li>
 *   <li>
 *     <b>Class:</b>
 *     If { @code true }, the role is allowed to add properties to the specified class.
 *   </li>
 *   <li>
 *     <b>Object:</b>
 *     Not applicable.
 *   </li>
 * </ol>
 *
 * @param canModifySchema { @code true } if the role is allowed to modify the schema of this resource.
 */
public void setCanModifySchema(boolean canModifySchema) {
    this.canModifySchema = canModifySchema;
}
}

```

1.3 retrofit 1.5.0

1.3.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

©2019 Cisco Systems, Inc. All rights reserved.