

Slack Library

# Table of Contents

API & Implementation .....	2
How to configure/use .....	3
Classpath .....	3
Configuration Properties .....	3
System Properties .....	3
Bootstrapping .....	3
Known issues.....	5
Dependencies .....	6

This module (`incode-module-slack-dom`) provides the ability to post to a Slack channel (with Slack running on the Cloud).

# API & Implementation

The `SlackService` defines the following API:

```
public class SlackService {  
    public boolean isConfigured() { ... } ①  
    public boolean channelExists(String channelName) { ... } ②  
  
    public void sendMessage( ③  
        String channelName,  
        String message) { ... }  
    public SlackMessageHandle<SlackMessageReply> sendMessage( ④  
        String channelName,  
        SlackPreparedMessage preparedMessage) { ... }  
}
```

- ① whether the service is available for use. If not configured, then the `sendMessage(...)` methods will silently no-op.
- ② allows consumers to check whether a channel exists (before attempting to use it in the `sendMessage(...)`). If the service is not configured, then returns `false`.
- ③ Sends a message to the specified channel. If no such channel exists, then an `IllegalArgumentException` will be thrown.
- ④ Allows more sophisticated messages to be sent (using the API of the underlying implementation library, [Simple Slack API](#)).

These actions are all programmatic and do not appear in the UI.

# How to configure/use

## Classpath

Update your classpath by adding this dependency in your `dom` project's `pom.xml`:

```
<dependency>
  <groupId>org.incode.module.slack</groupId>
  <artifactId>incode-module-slack-dom</artifactId>
  <version>1.16.1</version>
</dependency>
```

Check for later releases by searching [Maven Central Repo](#).

For instructions on how to use the latest `-SNAPSHOT`, see the [contributors guide](#).

## Configuration Properties

There is one mandatory configuration property.

```
isis.service.slack.authToken=xxxxx
```

The `authToken` can be obtained by logging into your [slack.com](#) account, on the settings page for bots. This token represents the credentials of the bot posting to the channel, so obviously should be kept private and not be shared around.

## System Properties

Optionally, an http/s proxy can be set, by setting two system properties:

```
-Dhttp.proxyPort=...
-Dhttp.proxyHost=...
```



Note that the system properties for SSL (`https.proxyHost` and `https.proxyPort`) are **NOT** checked.

## Bootstrapping

In the `AppManifest`, update its `getDependencies()` method, eg:

```
@Override
public Set<Module> getDependencies() {
    return Sets.newHashSet(
        ...
        new org.incode.module.slack.SlackModule(),
        ...
    );
}
```

# Known issues

None known at this time.

# Dependencies

Maven can report modules dependencies using:

```
mvn dependency:list -o -pl modules/lib/slack/impl -D excludeTransitive=true
```

which, excluding Apache Isis itself, returns these compile/runtime dependencies:

```
com.ullink.slack:simpleslackapi:jar:1.2.0
```