

# Research Platform Support For The CANARIE Registry And Monitoring System

canarie



## Introduction

As part of a long-term strategy to facilitate research platform software re-use, CANARIE requires that all platforms developed under the Research Software program support the CANARIE platform registry and platform monitoring services.

The platform registry allows potential users to discover and learn about deployed platforms. Along with basic platform location information (ie. a URL), the registry includes additional information about the platform provided by the creator. The goal is that this content will fully describe the details of the platform, how to use it, and how to obtain assistance.

CANARIE's platform monitoring service measures the availability and usage of research platforms.

The purpose of this document is to describe a web service interface that CANARIE requires developers to add to their platforms in order to support these services. The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119, regardless of case or capitalization (see <http://www.ietf.org/rfc/rfc2119.txt.pdf> for details).

## Methodology

The goal of this interface is to provide useful platform information in a consistent way while minimizing the impact on development. To support these goals, most of the features required in this document can be implemented by simple HTML pages. In situations where a web service call is required, JSON will be employed.

## Types of Platform

CANARIE supports two types of platform in the registry. These types are:

- Managed Platform – a single instance is deployed on the contributor's cloud or hardware resources.
- Self-Deployed Platform – a single demonstration instance is deployed on the contributor's cloud or hardware resources. Potential users can use this instance evaluation purposes. Once a user has decided to use this platform, they download some sort of installation media (source code, binaries, VM images, ...) from the contributor and deploy an instance on their own cloud or hardware resources.

The requirements for supporting the CANARIE registry and monitor vary between these types, so each has its own section in this document.

## Managed Platforms

The goal of the API enhancements described in this section is to provide useful platform information in a consistent way while minimizing the impact on existing platforms. To support these goals, simple HTML pages can be returned in many instances. In situations where a web service call is required, JSON will be employed.

To minimize the need for human maintenance of the registry, CANARIE would like all affected platforms to implement the URIs that are used by the registry and monitoring services in a standard subtree. Once your platforms's base URI, (referred to as <base> for the purposes of this document) is entered into the registry, all other items can be auto-populated.

## Authentication

As there is currently no common standard for user authentication among the platforms, the API described in this document must not require user authentication.

## Availability

CANARIE's monitoring service will poll all URIs listed below periodically. If any request fails or times out, the monitoring service will flag your platform as unavailable.

## Platform Info

URI - <base>/platform/info

Return basic identification and provenance information about the platform. CANARIE's monitoring service will poll this URI periodically.

When an HTTP GET is performed on this URI, and the Accept header specifies JSON (ie. application/json), the platform shall return the following:

```
{
  "name" : "<name of platform>",
  "synopsis": "<one or two sentences describing what the platform is for>",
  "version" : "<platform version identifier>",
  "institution": "<the name of the institution that contributed this platform>",
  "releaseTime": "<time at which this version of the platform was released>",
  "researchSubject": "<the research area to which this platform applies>",
```

```
    "supportEmail": "<email address for support in case of platform outage>",
    "tags": [<terms describing this platform – to be used in searches>]
}
```

where:

- The values of “name”, “synopsis”, “researchSubject” and “tags” may be entered either or both official languages.
- The value of “institution” will be taken as a string identifying the name of the organization that contributed the platform.
- The value of “version” will be taken as a string and can follow any versioning scheme you deem appropriate.
- “releaseTime” must be formatted as YYYY-MM-DDThh:mm:ssZ (ISO 8601) and must be expressed in UTC rather than in local time.
- “researchSubject” is the area of research that your platform applies to. You must use the research subject names published by NSERC. These can be found at [http://www.nserc-crsng.gc.ca/help-aide/codes-listedecodes\\_eng.asp](http://www.nserc-crsng.gc.ca/help-aide/codes-listedecodes_eng.asp). Use the subject name rather than the subject number and feel free to use the top-level subject names if none of the detailed subject names match the purpose of your platform. For platforms that are applicable to multiple areas of research, return “Multi-discipline”.
- “supportEmail” is an email address that users of your platform can use to communicate with you when the platform is unavailable. In most cases, users will get support through your platform support URI (see below) but in cases where the platform support URI is not working, users must be able to contact your support staff via email at this address. CANARIE will publish this address on the Research Middleware portal.
- “tags” shall contains a list of terms describing the platform, to aid in searches. This structure shall be implemented as a JSON array with strings as elements - for example:

```
    "tags": ["astronomy", "FITS"]
```

The content type for this response should specify “application/json”.

If an HTTP GET is performed and the Accept header does not indicate JSON, or there is no Accept header, platforms shall return an HTML page providing the information listed above in human-readable format. The content type for this response should specify “text/html”.

The HTTP HEAD request must also be supported when the Accept header does not indicate JSON.

## Platform Statistics

URI - <base>/platform/stats

Return information about the usage of your platform. To aid in determining availability, your platform must be designed such that requests to this URI will fail if other platform functionality is not available. In situations where your platform is not fully functional (if a database engine or other software service on which it relies is not available, for example), your platform must return an HTTP error 503 in response to a GET on this URI.

Instead of checking all subsystems every time this request is made, you may choose to implement a separate, periodically scheduled “self test” within your platform and base your response to a poll of the <base>/platform/stats URI on the most recent results of this test.

The research middleware ecosystem includes a number of different types of Managed Platforms and there is not necessarily a single usage measurement that is applicable to all of them. Instead, CANARIE is asking the platform creators to determine what usage metric makes sense for their particular platform. As such, the protocol described in this section allows the service creator to define their own <usage type> field for the status report. The guidelines are:

1. You must include one <usage type> field in the response packet defined in this section.
2. Field names must be meaningful and should convey units when they are not obvious. Consider names such as “CPU Hours”, “Work Packages” or “Logins”.

When an HTTP GET is performed on this URI, and the Accept header specifies JSON (ie. application/json) platforms shall return the following:

```
{
  "<usage type>" : "<service usage count since last reset>",
  "lastReset": "<the time and date at which the <usage type> field was last reset to zero>",
}
```

where:

- <usage type> is a meaningful field name that indicates the type of usage, along with units where appropriate, being reported (see above).
- The value of “<usage type>” is a positive integer. This value shall start at zero when the platform is first deployed and shall be incremented every time the platform is used, according to whatever usage criteria the service creator has defined.
- “lastReset” shall be formatted as YYYY-MM-DDThh:mm:ssZ (ISO 8601) and shall be expressed in UTC rather than local time. If you reset the <usage type> value to zero or it wraps around, this value must be updated accordingly.

The content type for this response should specify “application/json”.

If an HTTP GET is performed and the Accept header does not indicate JSON, or there is no Accept header, platforms shall return an HTML page providing the information listed above in human-readable format. The content type for this response should specify “text/html”.

The HTTP HEAD request must also be supported when the Accept header does not indicate JSON.

## **Platform Documentation**

URI - <base>/platform/doc

Make the online documentation for your platform available.

When an HTTP GET is performed, the user documentation for your platform shall be returned in a human-readable format. This information must include:

- A detailed description of the platform and what it does
- Training materials explaining how to use the platform
- Platform policies and procedures, terms of use, privacy statements,
- Plugin architecture and/or API descriptions, if applicable

If the documentation is hosted elsewhere, an HTTP redirect may be returned in response to this request. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

## **Platform Release Notes**

URI - <base>/platform/releasenotes

Return release notes describing the current version of your platform.

When an HTTP GET is performed, the release notes for the current version of your platform shall be returned in a human-readable format. As a minimum, these notes must identify:

- Changes made from previous versions
- Any known issues
- Work-arounds for known issues (where applicable).

If the release notes are hosted elsewhere, an HTTP redirect may be returned in response to this request. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

## Platform Support

URI - <base>/platform/support

Provide users with information on how to get support for your platform.

When an HTTP GET is performed, instructions for users on how to get support for your platform shall be returned in human-readable format. Include help desk contact info, a link to any bug tracking systems and/or forums, etc.

If the platform support information is hosted elsewhere, an HTTP redirect may be returned in response to this request. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

## Platform Source Code

URI - <base>/platform/source

When an HTTP GET is performed, return link(s) to the source code for your platform. This is only applicable if you are making your source code publicly available. If the source code is hosted elsewhere, you may return an HTTP redirect in response to this request. If you are not providing access to your platform’s source code, return status code 204 (No Content) must be returned.

If you are providing source code, also consider including design documentation. If the source code is hosted elsewhere, an HTTP redirect may be returned in response to this request. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

## Platform Try Me

URI - <base>/platform/tryme

Allow users to try out your platform online.

When an HTTP GET is performed, this URI shall return a page that allows the user to try your platform. Ideally, this link will allow unregistered users to try your platform in a sandbox environment or demo mode. CANARIE realizes that it may be prohibitively expensive to add this capability to some platforms. If that is the case, you may provide videos demonstrating the use of your platform at this URI.

If the try me content is hosted elsewhere, an HTTP redirect may be returned in response to this request. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

## Platform Licence

URI - <base>/platform/licence

Allow users to view your platform's licencing/usage term.

When an HTTP GET is performed, a page that indicates the licences and usage terms/restrictions associated with your platform shall be returned in human-readable format. Licensing information for any third-party components your platform incorporates or makes use of must be included, as applicable.

If the licensing information is hosted elsewhere, an HTTP redirect may be returned in response to this request. This URI must also support HTTP HEAD requests. The content type for the response should specify "text/html".

## Platform Provenance

URI - <base>/platform/provenance

Allow users to view the software provenance process used when making a new release of your platform publicly available.

When an HTTP GET is performed, a page that indicates what criteria has to be met before a version of your platform is deployed shall be returned in human-readable format. This page must include, as a minimum:

- Information about who authorizes releases
- A description of the validation procedures that are completed prior to each release
- A list of documentation that is created as part of the release package
- A statement regarding your policy in dealing with upgrades/patches to any third-party software packages you may be using

If the provenance information is hosted elsewhere, an HTTP redirect may be returned in response to this request. This URI must also support HTTP HEAD requests. The content type for the response should specify "text/html".

## Platform Fact Sheet

URI - <base>/platform/factsheet

Allow users to view the fact sheet for your platform. A fact sheet is a high-level description of your platform targeted at the general public. The CANARIE External Relations department will be more than happy to assist in the development of your platform's fact sheet. Your platform's fact sheet must follow CANARIE's standard format.

When an HTTP GET is performed, your platform's fact sheet shall be returned. If the fact sheet is hosted elsewhere, you may return an HTTP redirect in response to this request. This URI must also support HTTP HEAD requests. The content type for the response should specify "text/html".

## Self-Deployed Platforms

There is no single instance of a Self-Deployed platform. Instead, it is deployed by each user on their own resources. The CANARIE registry does not poll this type of platform directly for availability and usage information but the platform's documentation links are verified periodically. If you are the creator of a Self-Deployed platform and you would like your demonstration instance monitored as a Managed Platform, please contact us at [support@science.canarie.ca](mailto:support@science.canarie.ca).

For Self-Deployed Platforms, you will be asked to enter the following information into the registry at [science.canarie.ca](http://science.canarie.ca) manually when your platform is registered:

- Name – a short, descriptive name for your platform
- Synopsis - a 2-3 sentence description of what your platform does, to complement the Name
- Version – uniquely identifies the current version of your platform. The value of "Version" will be taken as a string and can follow any versioning scheme you deem appropriate.
- Contributor - the name of the institution that contributed the platform
- Research Subject - the research area to which this platform applies. You must use the research subject names published by NSERC. These can be found at [http://www.nserc-crsng.gc.ca/help-aide/codes-listedecodes\\_eng.asp](http://www.nserc-crsng.gc.ca/help-aide/codes-listedecodes_eng.asp). Use the subject name rather than the subject number and feel free to use the top-level subject names if none of the detailed subject names match the purpose of your service. For platforms that are applicable to multiple areas of research, return "Multi-discipline".
- Support Email - email address for users to obtain technical support for your platform
- Tags – a list of terms describing this platform to aid registry searching
- Administrators – the email address(es) of person(s) responsible for managing your platform within the registry at [science.canarie.ca](http://science.canarie.ca). Specified addresses must be either from institutions that are part of the Canadian Access Federation (CAF) or must be registered with the CAF Guest Identity Provider. Contact [support@science.canarie.ca](mailto:support@science.canarie.ca) for assistance.

- Note: The values of “name”, “synopsis”, “researchSubject” and “tags” may be entered either or both official languages.

In addition to the static information listed above, you will be asked to provide the following documentation URIs. These URIs will be polled periodically by the registry to verify the availability of your platforms’s documentation. Redirects are permitted. Unlike Managed Platforms, the documentation URIs for Self-Deployed Service do not need to have a common base.

### **Platform Documentation**

When an HTTP GET is performed, the user documentation for your platform shall be returned in a human-readable format. This information must include:

- A detailed description of the platform and what it does
- Training materials explaining how to use the platform
- A complete description of the API, if any
- Sample code illustrating API usage, as appropriate

This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

### **Platform Release Notes**

Return release notes describing the current version of your platform.

When an HTTP GET is performed, the release notes for the current version of your platform shall be returned in a human-readable format. As a minimum, these notes must identify:

- Changes made from previous versions
- Any known issues
- Work-arounds for known issues (where applicable).

This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

### **Platform Support**

Provide users with information on how to get support for your platform.

When an HTTP GET is performed, instructions for users on how to get support for your platform shall be returned in human-readable format. Include help desk contact info, a link to any bug tracking systems and/or forums, etc. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

### **Platform Source Code**

When an HTTP GET is performed, return link(s) to the source code for your platform. This is only applicable if you are making your source code publicly available. In this case, this URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

If you are not providing access to your platform’s source code it is not necessary to provide a URL and it may be left blank in the registry.

If you are providing source code, also consider including design documentation.

### **Platform Try Me**

Allow users to try out your platform online.

When an HTTP GET is performed, this URI shall return a page that allows the user to try your platform, possibly with fixed input. CANARIE realizes that it may be prohibitively complex to add this capability to some platforms. If that is the case, you may provide videos demonstrating the use of your platform at this URI. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

### **Platform Licence**

Allow users to view your platform’s licencing/usage terms.

When an HTTP GET is performed, a page that indicates the licences and usage terms/restrictions associated with your platform shall be returned in human-readable format. Licensing information for any third-party components your platform incorporates or makes use of must be included, as applicable. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

### **Platform Provenance**

Allow users to view the software provenence process used when making a new release of your platform publicly available.

When an HTTP GET is performed, a page that indicates what criteria has to be met before a version of your service is deployed shall be returned in human-readable format. This page must include, as a minimum:

- Information about who authorizes releases
- A description of the validation procedures that are completed prior to each release
- A list of documentation that is created as part of the release package
- A statement regarding your policy in dealing with upgrades/patches to any third-party software packages you may be using

This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.

### **Platform Download**

When an HTTP GET is performed, a page that allow users to download the distribution package for your platform shall be returned in human-readable format. Distribution packages include tarballs, VM images, compiled binaries, etc. If source code is your distribution medium, this link should be the same as the “Platform Source Code” link described above. This URI must also support HTTP HEAD requests. The content type for the response should specify “text/html”.