

# Document d'information sur les services de recherche du Registre et du Système de surveillance de CANARIE

canarie



## Introduction

Dans le cadre d'une stratégie à long terme visant à favoriser la réutilisation des services logiciels de recherche, CANARIE demande que les services élaborés grâce à son programme Logiciels de recherche soutiennent le registre et le service de surveillance situé sur <https://science.canarie.ca>.

Le registre permet aux utilisateurs potentiels de découvrir les services de recherche que vous avez déployés et d'en apprendre davantage à leur sujet. L'objectif est de décrire en détail le service, la façon de s'en servir et comment obtenir de l'aide, si besoin est, afin que l'utilisateur vive une expérience uniforme en parcourant les différents services.

Ce document décrit une interface que CANARIE prie les développeurs d'ajouter à leurs services afin de soutenir les fonctionnalités précitées. Les mots-clés « DOIT », « NE DOIT PAS », « OBLIGATOIRE », « DEVRA », « NE DEVRA PAS », « DEVRAIT », « NE DEVRAIT PAS », « RECOMMANDÉ », « PEUT » et « OPTIONNELLE » employés dans ce document ont le sens que leur donne la spécification RFC 2119, peu importe la casse et l'usage ou pas de majuscules (lire <http://www.ietf.org/rfc/rfc2119.txt.pdf> pour en savoir plus).

## Types de services

Le registre de CANARIE accepte trois types de service.

- Service géré – instance unique du service déployée dans le nuage du contributeur ou avec ses ressources matérielles. L'utilisateur accède au service par le réseau, en recourant à une API de service Web.
- Service auto-déployé – instance de démonstration déployée dans le nuage du contributeur ou avec ses ressources matérielles. L'utilisateur potentiel (c.-à-d., les concepteurs de plateformes) peut s'en servir aux fins d'évaluation. S'il décide de l'adopter, il télécharge ensuite un moyen d'installation quelconque (code source, compilation binaire, image de MV, etc.) fourni par le contributeur et déploie une instance du service sur son propre nuage ou son équipement. À l'instar des services gérés, l'utilisateur accède au service auto-déployé par le réseau, grâce à une API de service Web.
- Service intégré – expression désignant les services logiciels intégrés à une plateforme de recherche et auxquels on n'accède pas par le réseau au moyen d'une API. Les fragments en Javascript servant à la visualisation et les images de MV en sont des exemples.

Les contraintes permettant de soutenir le registre et de surveiller l'usage des services de CANARIE diffèrent selon le type de service et sont donc examinées séparément dans ce document.

## Services gérés

### Méthode

Les améliorations à l'API décrites dans cette partie ont pour but de fournir des informations utiles et homogènes sur le service sans que les services existants s'en ressentent, ou très peu. Dans cette optique, de simples pages HTML suffiront dans de nombreux cas. Quand l'intervention d'un service Web s'impose, on privilégiera JSON.

Pour assurer un fonctionnement nécessitant aussi peu d'interventions humaines que possible, CANARIE souhaite que les URI employés par le registre et le système de surveillance soient placés dans un embranchement de l'arborescence ordinaire. Une fois l'URI de base du service (<base> dans ce document) saisi dans le registre, le peuplement des autres éléments s'effectuera automatiquement.

### Authentification

Puisqu'aucune norme générale ne régit cet aspect entre les services de recherche, l'API décrite ici ne nécessite pas l'authentification de l'utilisateur.

### Disponibilité

Le système de surveillance de CANARIE sondera périodiquement les URI énumérés plus bas. Lorsque l'exécution d'une requête échoue ou prend un temps excessif, le système marquera le service comme non disponible.

### Renseignements

URI - <base>/service/info

Fournit des renseignements de base sur la nature et l'origine du service. Le système de surveillance de CANARIE sondera périodiquement cet URI.

Quand l'URI reçoit la requête HTTP GET et que l'en-tête Accept spécifie JSON (à savoir, application/json), le service restituera les renseignements que voici :

```
{
  "name" : "<nom du service>",
  "synopsis" : "<une ou deux phrases décrivant ce que fait le service>",
  "version" : "<version du service>",
```

```

    “institution” : “<nom de l’institution qui a offert le service>”,
    “releaseTime” : “<date à laquelle la version du service a été diffusée>”,
    “researchSubject” : ”<domaine de recherche auquel s’applique le service>”,
    “supportEmail” : “<adresse courriel du soutien technique en cas de panne>”,
    “category” : “<type de service>”;
    “tags” : [<termes décrivant le service – on les utilisera pour la fonction
recherche>]
}

```

### Précisions

- Les valeurs de “name”, “synopsis”, “researchSubject” et “tags” peuvent être saisies en anglais, en français ou dans ces deux langues.
- La valeur de “institution” correspondra à une chaîne indiquant le nom de l’organisation qui a offert le service.
- La valeur de “version” correspondra à une chaîne, selon le système d’identification des versions que vous jugez approprié.
- “releaseTime” sera formaté YYYY-MM-DDThh:mm:ssZ (ISO 8601) et exprimé en temps universel coordonné (UTC) plutôt qu’en temps local.
- “researchSubject” correspond au domaine de recherche auquel s’applique le service. Vous devez utiliser les noms de domaine publiés par le CRSNG, disponibles sur [http://www.nserc-crsng.gc.ca/help-aide/codes-listedecodes\\_fra.asp](http://www.nserc-crsng.gc.ca/help-aide/codes-listedecodes_fra.asp). Employez le nom plutôt que le numéro et n’hésitez pas à choisir le nom principal si aucun des noms secondaires ne correspond à celui du domaine auquel est destiné le service. Pour les services qui s’appliquent à plusieurs domaines de recherche, indiquer “multidisciplinaire”.
- “supportEmail” correspond à l’adresse à laquelle l’utilisateur écrira pour communiquer avec vous quand le service n’est pas disponible. Dans la majorité des cas, l’utilisateur obtiendra de l’aide via l’UPI de soutien (voir plus bas), mais il peut arriver que celui-ci ne fonctionne pas. L’utilisateur pourra alors communiquer avec le service du soutien technique par cette adresse. CANARIE affichera cette dernière sur son portail Logiciels de recherche.
- “category” – choisir une des valeurs suivantes :
  - Sensor Management/Data Acquisition
  - Data Storage and Retrieval
  - Data Manipulation
  - Data Visualization
  - Resource/Cloud Management
  - Service Registration/Discovery
  - Workflow/Service Scheduling
  - User Management/Authentication
  - Other
- “tags” dresse la liste des termes qui décrivent le service pour en faciliter la découverte. Cette structure prendra la forme d’un ensemble JSON ayant des chaînes comme éléments, par exemple :

```
“tags”:[“astronomie”, “FITS”]
```

Le type de contenu pour cette réponse devrait préciser “application/json”.

Quand il y a exécution de la requête HTTP GET et que l’en-tête Accept ne spécifie pas JSON, ou s’il n’y a pas d’en-tête Accept, le service restituera une page HTML fournissant l’information qui précède dans un format que peut lire un être humain. Le type de contenu pour cette réponse devrait indiquer “text/html”.

La requête HTTP HEAD sera aussi exécutée quand l’en-tête Accept n’indique pas JSON.

## Statistiques

URI - <base>/service/stats

Fournit des informations sur l’usage du service. Pour déterminer plus facilement sa disponibilité, le service sera conçu de façon à ce que les requêtes reçues par cet URI échouent si certaines fonctionnalités ne sont pas disponibles. Quand il n’est pas totalement fonctionnel (par exemple, si le moteur d’une base de données ou un service logiciel dont dépend celle-ci n’est pas disponible), le service renverra l’erreur HTTP 503 en guise de réponse à la requête GET appliquée à l’URI.

Au lieu d’enclencher une vérification de tous les systèmes secondaires à la réception d’une telle requête, vous pourriez procéder à un test d’auto-vérification distinct à intervalles réguliers et fonder la réponse à une interrogation de l’URI <base>/service/stats sur les résultats les plus récents de ce test.

L’écosystème des logiciels de recherche se compose de services gérés de nature diverse. On ne pourra donc pas nécessairement appliquer le même étalon à l’utilisation de tous ces services. CANARIE demande plutôt au créateur d’établir quelle mesure revêt le plus de sens pour son service. Le protocole décrit dans cette section lui permet de définir son propre champ <type d’utilisation>, qui sera repris dans le rapport d’état. Voici les consignes à respecter.

1. Le paquet réponse décrit ici doit inclure un champ <type d’utilisation>.
2. Ce champ doit avoir un nom significatif et devrait préciser l’unité employée quand la chose n’est pas évidente. Songez à des noms comme “heures CPU” ou “ouvertures de fichiers”.
3. Une version antérieure de ce document mentionnait un champ obligatoire baptisé “invocations” : “<nombre de fois où le service a été utilisé depuis la dernière remise à zéro>”. Vous pouvez vous servir de cette définition pour <type d’utilisation>, s’il y a lieu.

Quand l’URI reçoit une requête HTTP GET et que l’en-tête Accept spécifie JSON (à savoir, application/json), le service restituera ce qui suit :

```
{
```

```
    “<type d’utilisation>” : “<nombre de fois où le service a été utilisé depuis la
dernière remise à zéro>”,
    “lastReset”: “<heure et date où le champ <type d’utilisation> a été remis à zéro
pour la dernière fois>”,
}
```

#### Précisions

- <type d’utilisation> est un nom de champ significatif se rapportant à la nature de l’usage que l’on fait du service, avec les unités employées, s’il y a lieu (voir plus haut).
- La valeur de “<type d’utilisation>” est un entier positif. Cette valeur correspond à zéro au déploiement du service et augmente d’une unité à chaque utilisation, selon le critère d’utilisation défini par le créateur. Attention : les accès à l’API décrite dans ce document ne devraient être comptabilisés au même titre qu’une utilisation normale du service.
- “lastReset” doit respecter le format YYYY-MM-DDThh:mm:ssZ (ISO 8601) et sera exprimé en temps universel coordonné (UTC) plutôt qu’en temps local. Lorsque la valeur de <type d’utilisation> est remise à zéro ou revient à sa valeur initiale, la valeur de “lastReset” doit être actualisée en conséquence.

Le type de contenu de la réponse devrait indiquer “application/json”.

Quand il y a exécution de la requête HTTP GET et que l’en-tête Accept n’indique pas JSON, ou s’il n’y a pas d’en-tête Accept, le service restituera une page HTML fournissant les informations décrites ci-dessus dans un format que peut lire l’être humain. Le type de contenu pour cette réponse devrait spécifier “text/html”.

La requête HTTP HEAD sera également exécutée quand l’en-tête Accept n’indique pas JSON.

## Documentation

URI - <base>/service/doc

Permet de consulter la documentation en ligne sur le service.

Quand il y a exécution de la requête HTTP GET, la documentation sur le service destinée à l’utilisateur est restituée dans un format que peut lire l’être humain. L’information en question comprendra ce qui suit :

- une description détaillée du service et de ce qu’il fait;
- un guide expliquant comment utiliser le service;
- une description complète de l’API;
- un échantillon de code illustrant l’utilisation de l’API, s’il y a lieu.

Si la documentation est hébergée ailleurs, la requête pourrait donner lieu à un réacheminement HTTP. Cet URI acceptera aussi les requêtes HTTP HEAD. Le type de contenu de la réponse devrait indiquer “text/html”.

## Notes de mise à jour

URI - <base>/service/releasenotes

Fournit des notes décrivant la version la plus récente du service.

Quand il y a exécution de la requête HTTP GET, l'utilisateur reçoit les notes de mise à jour sur la version actuelle du service dans un format que peut lire l'être humain. Ces notes préciseront au moins ce qui suit :

- les modifications apportées aux versions antérieures;
- les problèmes connus;
- les façons de contourner les problèmes connus (s'il y a lieu).

Si les notes de mise à jour sont hébergées ailleurs, la requête pourrait donner lieu à un réacheminement HTTP. Cet URI acceptera aussi les requêtes HTTP HEAD. Le type de contenu de la réponse devrait indiquer “text/html”.

## Soutien

URI - <base>/service/support

Explique à l'utilisateur comment obtenir de l'aide sur le service.

Quand il y a exécution de la requête HTTP GET, l'utilisateur reçoit des instructions lui expliquant comment obtenir de l'aide dans un format que peut lire l'être humain. Ces renseignements comprennent les coordonnées du service de soutien technique, un lien menant aux systèmes éventuels de dépistage des bogues ou aux forums, etc.

Si l'information sur le soutien technique est hébergée ailleurs, la requête pourrait donner lieu à un réacheminement HTTP. Cet URI acceptera aussi les requêtes HTTP HEAD. Le type de contenu de la réponse devrait indiquer “text/html”.

## Code source

URI - <base>/service/source

Quand il y a exécution de la requête HTTP GET, l'utilisateur obtient le ou les liens menant vers le code source du service. Ceci ne s'applique que si vous mettez le code source à la disposition du public. Si le code source est hébergé ailleurs, la demande pourrait donner lieu à un réacheminement HTTP. Si vous n'autorisez pas l'accès au code source, la réponse consistera en un code d'état 204 (aucun contenu).

Si vous donnez accès au code source, songez à inclure de la documentation sur sa conception. Si le code source est hébergé ailleurs, la requête pourrait donner lieu à un réacheminement HTTP. L'URI acceptera aussi les requêtes HTTP HEAD. Le type de contenu de la réponse devrait indiquer "text/html".

## Essayez-moi

URI - <base>/service/tryme

Permet à l'utilisateur d'essayer le service en ligne.

Quand il y a exécution de la requête HTTP GET, cet URI restitue une page permettant à l'utilisateur de tester le service, peut-être avec des valeurs fixes. CANARIE est conscient qu'ajouter une telle capacité à certains services pourrait s'avérer trop complexe. Dans un tel cas, l'URI pourrait donner accès à des bandes vidéo illustrant le fonctionnement du service.

Si le contenu « essayez-moi » est hébergé ailleurs, la requête pourrait donner lieu à un réacheminement HTTP. Cet URI acceptera aussi les requêtes HTTP HEAD. Le type de contenu de la réponse devrait indiquer "text/html".

## Licence d'exploitation

URI - <base>/service/licence

Permet à l'utilisateur de consulter la licence ou les conditions d'exploitation associées au service.

Quand il y a exécution de la requête HTTP GET, l'utilisateur reçoit une page indiquant les licences d'exploitation et les conditions/restrictions associées à l'usage du service dans un format que peut lire l'être humain. Le cas échéant, on inclura l'information sur les licences d'exploitation qui s'appliquent aux composants des tierces parties intégrés au service.

Si l'information sur la licence d'exploitation est hébergée ailleurs, la requête pourrait donner lieu à un réacheminement HTTP. Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer "text/html".

## Origine

URI - <base>/service/provenance

Permet à l'utilisateur d'établir l'origine du logiciel quand une nouvelle version du service est rendue publique.

Quand il y a exécution de la requête HTTP GET, l'utilisateur obtient une page dans un format que peut lire l'être humain indiquant les critères qui doivent être respectés avant le déploiement du service. Cette page fournira au moins ce qui suit :

- des renseignements sur la personne qui autorise la diffusion;
- une description du processus de validation précédant chaque diffusion;
- la liste de documents créée lors de la diffusion;
- une déclaration expliquant la politique relative aux mises à niveau et aux rustines qui s'appliquent aux logiciels de tiers que le service pourrait intégrer.

Si l'information sur l'origine est hébergée ailleurs, la requête pourrait donner lieu à un réacheminement HTTP. Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer "text/html".

## Services auto-déployés

Puisque le fonctionnement d'aucune plateforme de recherche ne dépend d'un seul service auto-déployé, le registre de CANARIE ne sonde pas directement les services de ce genre pour en vérifier la disponibilité ou recueillir des données sur leur utilisation. Cependant on vérifie périodiquement les liens conduisant à la documentation qui s'y rapporte. Si vous avez créé un service auto-déployé et aimeriez que son instance de démonstration soit surveillée comme s'il s'agissait d'un service géré, veuillez écrire à

[support@science.canarie.ca](mailto:support@science.canarie.ca).

Vous devrez saisir manuellement les renseignements qui suivent dans le registre de science.canarie.ca à l'inscription du service auto-déployé.

- Nom – court intitulé décrivant le service.
- Synopsis - description en deux ou trois phrases de ce que fait le service, en complément au nom.
- Version – identifie de façon unique la version actuelle du service; la valeur de "Version" correspondra à une chaîne, selon le système d'identification des versions que vous jugez approprié.
- Contributeur – nom de l'institution qui a offert le service.

- Sujet de recherche - correspond au domaine de recherche auquel s'applique le service. Vous devez utiliser les noms de domaine publiés par le CRSNG, disponibles sur [http://www.nserc-crsng.gc.ca/help-aide/codes-listedecodes\\_fra.asp](http://www.nserc-crsng.gc.ca/help-aide/codes-listedecodes_fra.asp). Employez le nom plutôt que le numéro et n'hésitez pas à choisir le nom principal si aucun des noms secondaires ne correspond à celui du domaine auquel est destiné le service. Pour les plateformes qui s'appliquent à plusieurs domaines de recherche, indiquer "multidisciplinaire".
- Courriel de support – adresse à laquelle l'utilisateur peut écrire pour obtenir un soutien technique.
- "category" – nature du service (choisir parmi celles qui suivent) :
  - Sensor Management/Data Acquisition
  - Data Storage and Retrieval
  - Data Manipulation
  - Data Visualization
  - Resource/Cloud Management
  - Service Registration/Discovery
  - Workflow/Service Scheduling
  - User Management/Authentication
  - Other
- Marqueurs – liste de termes descriptifs sur le service qui faciliteront les recherches dans le registre.
- Administrateurs – adresse courriel de la ou des personnes qui gèrent le service dans le registre de science.canarie.ca. Il peut s'agir d'une institution faisant partie de la Fédération canadienne d'accès (FCA) ou d'un fournisseur d'identité inscrit comme invité à la FCA. Pour obtenir de l'aide, écrire à [support@science.canarie.ca](mailto:support@science.canarie.ca).
- Remarque : les valeurs de "Nom", "Synopsis", "Sujet de recherche" et "Marqueurs" peuvent être saisies en anglais, en français ou dans ces deux langues.

En plus des informations statiques qui précèdent, vous serez priés de fournir l'URI de la documentation qui suit. Le registre vérifiera périodiquement si la documentation en question est disponible. Les réacheminements sont autorisés. À l'inverse des services gérés, les URI renvoyant à la documentation des services auto-déployés n'ont pas besoin de posséder un tronc commun.

## Documentation

Quand il y a exécution de la requête HTTP GET, la documentation sur le service destinée à l'utilisateur est restituée dans un format que peut lire l'être humain. L'information en question comprendra ce qui suit :

- une description détaillée du service et de ce qu'il fait;
- un guide expliquant comment utiliser le service;
- une description complète des API, s'il y en a;
- un échantillon de code illustrant le fonctionnement des API, au besoin.

Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer “text/html”.

### **Notes de mise à jour**

Fournit des notes décrivant la version la plus récente du service.

Quand il y a exécution de la requête HTTP GET, l'utilisateur obtient les notes de mise à jour sur la version actuelle du service dans un format que peut lire l'être humain. Ces notes préciseront au moins ce qui suit :

- les modifications apportées aux versions antérieures;
- les problèmes connus;
- les façons de contourner les problèmes connus (s'il y a lieu).

Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer “text/html”.

### **Soutien**

Explique à l'utilisateur comment obtenir de l'aide sur le service.

Quand il y a exécution de la requête HTTP GET, l'utilisateur reçoit des instructions lui expliquant comment obtenir de l'aide dans un format que peut lire l'être humain. Ces renseignements comprennent les coordonnées du service de soutien technique, un lien menant aux systèmes éventuels de dépiage des bogues ou aux forums, etc. Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer “text/html”.

### **Code source**

Quand il y a exécution de la requête HTTP GET, retourne le ou les liens menant au code source du service. Ceci ne s'applique que si vous mettez le code source à la disposition du public. Dans un tel cas, l'URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer “text/html”.

Si vous n'autorisez pas l'accès au code source de le service, il est inutile de fournir un URI. Ne rien écrire dans l'espace prévu du registre.

Si vous donnez accès au code source, songez également à fournir la documentation sur sa conception.

### **Essayez-moi**

Permet à l'utilisateur d'essayer le service en ligne.

Quand il y a exécution de la requête HTTP GET, cet URI restitue une page permettant à l'utilisateur de tester le service, éventuellement avec des valeurs fixes. CANARIE est conscient qu'ajouter une telle capacité à certains services pourrait s'avérer trop complexe.

Dans un tel cas, l'URI pourrait donner accès à des bandes vidéo illustrant le fonctionnement du service. Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer "text/html".

### **Licence d'exploitation**

Permet à l'utilisateur de consulter la licence ou les conditions d'exploitation associées au service.

Quand il y a exécution de la requête HTTP GET, l'utilisateur reçoit une page dans un format que peut lire l'être humain indiquant les licences d'exploitation et les conditions/restrictions associées à l'usage du service. Le cas échéant, on inclura l'information sur les licences d'exploitation qui s'appliquent aux composants des tierces parties intégrés au service. Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer "text/html".

### **Origine**

Permet à l'utilisateur d'établir l'origine du logiciel quand une nouvelle version du service est rendue publique.

Quand il y a exécution de la requête HTTP GET, l'utilisateur obtient une page dans un format que peut lire l'être humain indiquant les critères qui doivent être respectés avant le déploiement du service. Cette page indiquera au moins ce qui suit :

- des renseignements sur la personne qui autorise la diffusion;
- une description du processus de validation précédant chaque diffusion;
- la liste de documents créée lors de la diffusion;
- une déclaration expliquant la politique relative aux mises à niveau et aux rustines qui s'appliquent aux logiciels de tiers que le service pourrait intégrer.

Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait indiquer "text/html".

### **Téléchargement**

Quand il y a exécution de la requête HTTP GET, l'utilisateur accède à une page dans un format que peut lire l'être humain lui permettant de télécharger la trousse de diffusion du service. Cette trousse inclura les archives TAR, des images de MV, des compilations de codes binaires, etc. Si le code source sert de moyen de diffusion, ce lien pourrait être le même que celui du code source décrit plus haut. Cet URI acceptera aussi la requête HTTP HEAD. Le type de contenu de la réponse devrait être "text/html".

### **Services intégrés**

Pour l'instant, les exigences applicables aux services intégrés sont identiques à celles des services auto-déployés.