|  |
| --- |
| 統計等情報流通連携基盤システム  システム概要書  株式会社横須賀テレコムリサーチパーク |

目次

[1 はじめに 3](#_Toc387256422)

[1.1 統計等情報流通連携基盤システムの概要 3](#_Toc387256423)

[1.2 本書の構成 4](#_Toc387256424)

[2 統計等情報流通連携基盤システムの構成 5](#_Toc387256425)

[2.1 Frontend Server 6](#_Toc387256426)

[2.2 Developer Site Server 6](#_Toc387256427)

[2.3 User Information Server 7](#_Toc387256428)

[2.4 Graph Database Server 7](#_Toc387256429)

[3 保守・運用項目 8](#_Toc387256430)

[3.1 運用 8](#_Toc387256431)

[3.2 保守 9](#_Toc387256432)

[4 SLAを満たすためのシステム構成とコスト 11](#_Toc387256433)

[4.1 サービス稼働率目標値の達成に向けた対策 11](#_Toc387256434)

[4.1.1 ハードウェア障害・ソフトウェア障害対策 11](#_Toc387256435)

[4.1.2 ハードウェア陳腐化対策 11](#_Toc387256436)

[4.2 オンライン応答時間遵守率の達成に向けた対策 11](#_Toc387256437)

[4.2.1 アクセス数増加の対策 11](#_Toc387256438)

[4.3 セキュリティ上の重大障害対策 12](#_Toc387256439)

[4.3.1 不要なサービスの停止 12](#_Toc387256440)

[4.3.2 不要なポートの閉鎖 12](#_Toc387256441)

[4.3.3 鍵認証によるSSHログイン 12](#_Toc387256442)

[4.3.4 セキュリティパッチの適用 12](#_Toc387256443)

[4.4 システム運用上の重大障害対策 12](#_Toc387256444)

[4.5 運用コスト試算 13](#_Toc387256445)

[付録　統計等情報流通連携基盤システムのSLA 14](#_Toc387256446)

# はじめに

本書は、「情報流通連携基盤の統計情報・データカタログへの適用実証」（以下「本実証と呼ぶ」）において構築した統計等情報流通基盤システム（以下「本システム」と呼ぶ）の概要と、運用・保守に必要な項目を示す。

## 統計等情報流通連携基盤システムの概要

本システムは、情報流通連携基盤システム外部仕様書[[1]](#footnote-1)（以下「外部仕様書」と呼ぶ）が規定するSPARQL-Based Commandにより、アプリケーションからのRDFデータに対する検索・取得要求を受け付ける「API機能」と、開発者を支援するためのAPIドキュメント、サンプルプログラム、ライブラリ等を提供する「開発者サイト機能」からなるシステムである（図 1）。なおAPI機能には、次世代統計システム利用API[[2]](#footnote-2)やデータカタログAPI[[3]](#footnote-3)から取得した情報をRDF化した結果を保持する機能を含む。

本システムが提供するデータは以下の通りである。

* 統計センターの「次世代統計利用システム[[4]](#footnote-4)」が提供する統計情報のうち以下のもの
  + 次世代統計利用システムAPIから取得する統計表・関連情報
    - メッシュ統計、国勢調査小地域集計
    - 平成12年から現在までの以下のデータを収録
      * 国勢調査（500mメッシュ、1kmメッシュ、小地域）
    - 統計表は都道府県毎に分割されているため、およそ2,300件存在し、含まれるデータ数はおよそ1億1千万件である
  + HTMLやExcel等のデータから抽出する統計表・関連情報
    - 日本標準産業分類の一覧
      * 平成5年に改正されてから現在までの分類（計3回）
        + HTMLで公開されている範囲を対象としている
    - 日本標準職業分類の一覧
      * 平成9年に改正されてから現在までの分類（計2回）
        + HTMLで公開されている範囲を対象としている
* 内閣官房が提供する「政府オープンデータカタログサイト試行版[[5]](#footnote-5)」が提供するデータカタログのメタデータ

これらのデータの総サイズは46GB、 RDFトリプル総数は7.7億件であった。



図 1 統計等情報流通連携基盤システムの位置づけ

なお、「API機能」は、アプリケーションに対してデータの検索・取得機能を提供する機能であり、ユーザインタフェースを持たない。一方「開発者サイト機能」は、開発者に対してAPIドキュメント、サンプルプログラム、ライブラリ等を提供する機能であり、Webページとして構築した。

## 本書の構成

本書の構成は、以下の通りである。

* 第2章では、本システムの機能構成と各機能を提供するサーバのスペックを示す。
* 第3章では、本システムの運用・保守において必要な項目を示す。
* 第4章では、本システムがSLA(Service Level Agreement)を満たすためのサーバ追加に関するシステム構成や、その際に必要となるコストを示す。

# 統計等情報流通連携基盤システムの構成

本章では、統計等情報流通連携基盤システムの構成と、構成要素である各サーバのソフトウェア構成を示す。

本システムは、以下の4つのサーバから構成される。システムの構成図を図 2に示す。また、1分間に100件程度のリクエストを受け付け、それぞれのリクエストへの応答時間を1秒程度とするためには、それぞれのサーバのスペックを表 1のように設定することが望ましい。ただし、この値は本実証で扱ったデータと実証中に受け付けたリクエストから算出したものであり、応答時間は受付クエリの複雑度に依存することに注意が必要である。

* Frontend Server（api.odstat.jp）
  + 外部のアプリケーションに対して、情報流通連携基盤システム外部仕様書が規定するAPI（Application Programming Interface）を提供する。
* Developer Site Server（developer.odstat.jp）
  + 開発者サイトをホスティングする。
* User Information Server
  + 開発者のユーザ情報を保持する。
* Graph Database Server
  + RDFデータを格納し、SPARQLで検索する機能を提供する。



図 2 システム構成図

表 1 サーバスペック一覧

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Frontend Server | Developer Site Server | User Information Server | Graph Database Server |
| CPU | Intel Xeon 1.7GHz相当 1コア | Intel Xeon 1.7GHz相当 1コア | Intel Xeon 1.7GHz相当 2コア | Intel Xeon 1.7GHz相当 8コア |
| Memory | 1.7GiB | 1.7GiB | 6.5GiB | 68GiB |
| Storage | 8GiB | 8GiB | 60GiB | 120GiB |

以下、それぞれのサーバのソフトウェア環境を示す。

## Frontend Server

Frontend Serverは、情報流通連携基盤システム外部仕様書が規定する、SPARQL-Based CommandのAPIを提供する。本実証では、表 2の環境でFrontend serverを構築した。

表 2 Frontend Server ソフトウェア環境

|  |  |
| --- | --- |
|  | ソフトウェア名・バージョン |
| Operating System | Amazon Linux AMI release 2013.09 |
| リバースプロキシ | nginx 1.4.3 |
| Ruby実行環境 | Ruby 2.1.0p0 |

## Developer Site Server

Developer Site Serverは、情報流通連携基盤の開発者サイトをホスティングする。本実証では、表 3の環境でDeveloper Site Serverを構築した。

表 3 Developer Site Server ソフトウェア環境

|  |  |
| --- | --- |
|  | ソフトウェア名・バージョン |
| Operating System | Amazon Linux AMI release 2013.09 |
| リバースプロキシ | nginx 1.4.3 |
| Database | PostgreSQL 9.3.1 |
| Ruby実行環境 | Ruby 2.0.0p247 |

## User Information Server

User Information Serverは、情報流通連携基盤の開発者サイトに登録されたユーザ情報を管理する。本実証では、表 4の環境でUser Information Serverを構築した。

表 4 User Information Server ソフトウェア環境

|  |  |
| --- | --- |
|  | ソフトウェア名・バージョン |
| Operating System | Amazon Linux AMI release 2013.09 |
| Database | MongoDB 2.4.9 |

## Graph Database Server

Graph Database Serverは、RDFで記述されたデータを格納し、情報流通連携基盤のSPARQL 1.1準拠APIで検索するための機能を提供する。本実証では、表 5の環境で構築を行った。

なお、この環境でGraph Database Serverを構築する場合，Graph Database Systemのインデクス用に1トリプルあたり39バイト必要である。従って、ディスク容量は、格納するデータ本体に加えて、トリプル数×39バイト以上必要である。

表 5 Graph Database Server ソフトウェア環境

|  |  |
| --- | --- |
|  | ソフトウェア名・バージョン |
| Operating System | Amazon Linux AMI release 2013.09 |
| Graph Database System | Virtuoso Open-Source Edition 6.1.8 |

# 保守・運用項目

本システムのAPI機能、開発者サイト機能について、必要となる保守・運用項目を示す。いずれも保守・運用項目は同一であるため、両機能に適用すべき項目をまとめて記述する。

## 運用

本システムを安定稼働するために、以下の項目を実施する。

#### セキュリティパッチの適用

システムのセキュリティを確保するため、セキュリティパッチが公開された場合にはセキュリティパッチを適用し、システムを常に最新の状態に保つ。本実証では、脆弱性データベースであるJVN iPedia[[6]](#footnote-6)を確認し、利用しているソフトウェアに対する深刻度が「警告」以上の脆弱性に対してセキュリティパッチが公開された場合はセキュリティパッチを1週間以内に適用するという運用を行った。

#### 修正プログラムの適用

本システムを構成するモジュールに修正プログラムがある場合、または本システムが利用しているソフトウェア（ruby、Virtuoso、Mongo等）に修正プログラムがある場合には、これを適用してシステムを最新の状態に保つ。本実証では、システム運用開始時点において最新のバージョンのソフトウェアをPackage Management System等を利用してインストールし、運用を行った。Virtuosoに関してはOSのバージョンとの相性もあるため、ソースコードからインストールし、バージョンを据え置いて運用を行った。

#### システム監視

システムの品質を管理するために、システムの監視を行う。またインシデント発生時には、インシデントに対応する。本実証では、APIサーバと開発者サイトのnginxのログファイルを確認し、利用者のリクエストに対するHTTPのレスポンスコードが200（正常終了）となっているかどうかを1日2回以上確認すること、および1日1回、APIリクエストの結果が正しく取得できることを確認することで監視を行った。

#### ヘルプデスク業務

開発者サイト等から寄せられるシステム利用者の対応を行う。また必要に応じて開発者サイトのドキュメントの更新作業や、ニュースとして開発者サイトに掲載することとする。本実証では、毎日開発者サイトのフォーラムを確認した。また、システムのアップデートなどを行った場合には、ニュースとして開発者サイトに掲載した。

#### 可用性の管理

電源の管理などを行うことで、システムが安定的に運用できるようにする。また、異常検出時には適宜対応を行う。本項目には、システムで動作するソフトウェアの管理だけではなく、物理サーバや仮想サーバの管理を含む。なおクラウドサービス等を活用して物理サーバの管理を第三者に依頼することも可能とする。本実証では、クラウドサービスを活用して、サーバの物理的なトラブルの影響を受けづらくしたため、物理的な要因等によるサーバのトラブルなく運用することができた。

#### サービスサポート

システム監視やヘルプデスクで挙げられた問題を一元管理して、過去のインシデント情報を活用してサービスの向上に努める。本実証では、システムの監視結果や問題点をナレッジベースで管理を行い、また担当者が一元的に把握を行うことで、サービスの向上に努めた。

#### 問題管理

サービスサポートにて挙げられた問題を管理し、優先度付けを行った上で、優先度順に問題の原因を特定する。早急に解決する必要があり、なおかつ解決に時間の必要な問題には、一時的な解決をまず行い、その後、恒久的な解決策の策定を行う。本実証においても、本手順に基づいて問題の対応を行った。

## 保守

本システムを24時間365日稼働するために、以下の保守業務を行う。

#### 修正プログラムの調査・作成

本システムに不具合がある場合、または利用しているツール等に不具合やセキュリティ上の欠陥がある場合、適切なパッチの調査を行う。適切なパッチが存在しない場合には、パッチの作成等を含めた適切な対応を検討する。本実証では、1週間に1度程度、最新のパッチの存在有無を確認した。パッチが存在する場合には、運用ポリシーに則って適用を行った。

#### ハードウェア保守

ハードウェアが原因のシステムトラブルがあった場合には、障害箇所の切り分けや原因調査を行い、ハードウェアの交換、修理等を速やかに行う。本実証ではクラウドサーバを活用したため、ハードウェア保守は発生しなかった。

# SLAを満たすためのシステム構成とコスト

本システムが付録に示すSLA(Service Level Agreement)を満たすためのサーバ追加に関するシステム構成や、その際に必要となるコストを記述する。

## サービス稼働率目標値の達成に向けた対策

### ハードウェア障害・ソフトウェア障害対策

ハードウェア障害対策のため、サーバの仮想化を行う。また、1日1回程度、マシンイメージのバックアップを取得する。サービス稼働率を保つため、サーバの監視を行い、サーバのトラブルを検出する。仮想サーバにすることで、サーバの応答がなくなる等サーバにトラブルが生じた場合は、他の物理サーバ上でバックアップを利用して直ちにサービス運用を開始できる。本実証では、仮想マシン全体となる200GiBのマシンイメージをバックアップし、1日前の仮想マシンの状態をバックアップした。常に上書きバックアップとしているため、200GiB分の保存領域を必要とした。

### ハードウェア陳腐化対策

サーバは仮想サーバを利用することとして、物理サーバの陳腐化に仮想サーバが影響されないように配慮を行うこととする。また、仮想サーバとしたことにより、物理サーバにトラブルが生じた場合でも即座に別のサーバに仮想サーバを移設して稼働させることができる。

仮想サーバは、クラウド事業者の提供するVPS（Virtual Private Server）サービスを活用することで、物理的なサーバの設置場所の災害等による影響を受けにくくする。本実証ではクラウドサーバとしてAmazon Web Serviceを活用してサーバの運用を行った。このため、ハードウェアの管理はクラウド事業者任せにでき、さらにハードウェアが陳腐化してもすぐに別の物理サーバへ移設できることから、ハードウェア陳腐化の影響を受けないような構成とすることができた。

## オンライン応答時間遵守率の達成に向けた対策

### アクセス数増加の対策

アクセス数が増加した際には、単位時間当たりのSPARQLの検索クエリ数を多くするためにはGraph Database Serverを、開発者サイトの応答時間遵守率を保つためにはDeveloper Site Serverを追加することで対応可能である。本実証では、Developer Site ServerやAPI Server、Graph Database Serverなど、サーバの役割を単機能としたため、アクセス数が増加して応答時間を向上させたいサービス毎に追加・性能向上させるべきサーバが明確であり、アクセス数増加に対応しやすいサーバ構成にできた。

## セキュリティ上の重大障害対策

### 不要なサービスの停止

不要なサービスを停止することで、セキュリティ上のインシデントの発生する要因を減らす。本実証で利用したサービスは、Webサーバ（HTTP、HTTPS）、メールサーバ（SMTP）、Graph DBサーバソフトウェア、APIサーバソフトウェア、開発者サイトソフトウェア、データベースサーバと、メンテナンス用のSSHサーバである。

### 不要なポートの閉鎖

利用しないポートをFirewallによって閉鎖することで、攻撃対象となる要因を減らす。本実証ですべてのIPアドレスから接続を許可したポートは、HTTP（開発者サイトではセキュリティ向上のためHTTPSへリダイレクト）、HTTPSのみである。

### 鍵認証によるSSHログイン

サーバへのSSHログインを鍵認証にすることで、サーバの乗っ取られないように対策した。

### セキュリティパッチの適用

セキュリティパッチの適用漏れを防ぐため、可能な限りOSの提供しているPackage Management Systemを利用してアプリケーションをインストールする。本実証では、Graph DatabaseのVirtuoso以外はPackage Management System等を利用してインストールを行い、最新版を取得しやすいように環境を構築した。

## システム運用上の重大障害対策

長期にわたって正常に稼働できない状況を防ぐための対策は、4.1節に記載したサービス稼働率目標値の達成に向けた対策と同一である。

## 運用コスト試算

上記のSLAを満たすためのコストを、実装詳細仕様書に記載した構成でクラウドサーバの一つであるAmazon Web Serviceを利用した場合を例に試算する。なお本書の表 1（p. 6）に示した構成で、SLAを満たすことが可能である。その場合のコストは、以下の通りである。

* Frontend Server (m1.small, EBS 8GB, Snapshot 40GB)
  + 月額$69.10
* Developer Site Server (m1.small, EBS 8GB, Snapshot 40GB)
  + 月額$69.10
* User Information Server (m1.xlarge, EBS 60GB, Snapshot 300GB)
  + 月額$291.30
* Graph Database Server (m2.4xlarge, EBS 120GB, Snapshot 600GB)
  + 月額$1548.84

合計で、月額$1978.34となる。

# 付録　統計等情報流通連携基盤システムのSLA

統計等情報流通連携基盤システムのSLAを、以下のように定める。

1. 情報流通連携基盤運用に係る業務
   1. サービス稼働率  
      システムの稼働率は95%以上とする。なお稼働率は以下の計算式で計算する。  
      稼働率(%)={1-(1年間の停止時間)÷1年間の稼働予定時間}  
      ※1年間の稼働予定時間=(24時間×1年の日数)-計画メンテナンス等による停止時間
   2. セキュリティ上の重大障害  
      システムに第三者がログインできないこと。
   3. システム運用上の重大障害の件数  
      長期にわたり正常に稼働できないような重大障害の件数は0件であること。
   4. セキュリティパッチの適用  
      脆弱性対策情報データベースに掲載されている、深刻度が危険である脆弱性に対するセキュリティパッチを1週間以内に適用すること。
2. 開発者サイト運用に係る業務
   1. サービス稼働率  
      システムの稼働率は97%以上とする。稼働率の計算式は、情報流通連携基盤と同一である。
   2. オンライン応答時間遵守率  
      システムの応答時間は1秒以内とし、オンライン応答時間遵守率は80%以上とする。
   3. セキュリティ上の重大障害  
      開発者サイトに登録されたユーザの個人情報の漏洩件数は0件であること。
   4. システム運用上の重大障害の件数  
      長期にわたり正常に稼働できないような重大障害の件数は0件であること。
   5. セキュリティパッチの適用  
      脆弱性対策情報データベースに掲載されている、深刻度が危険である脆弱性に対するセキュリティパッチを1週間以内に適用すること。

1. http://www.opendata.gr.jp/cfc/。ただし、SPARQL-Based Commandの内部で、平成25年度版仕様にあるSecurity Management Commandを利用している。 [↑](#footnote-ref-1)
2. http://statdb.nstac.go.jp/system-info/api/ [↑](#footnote-ref-2)
3. http://docs.ckan.org/en/latest/api/ [↑](#footnote-ref-3)
4. http://statdb.nstac.go.jp/ [↑](#footnote-ref-4)
5. http://data.go.jp/ [↑](#footnote-ref-5)
6. http://jvndb.jvn.jp/ [↑](#footnote-ref-6)