Engineer

QGIS

無償で使えるオープンソースのデスクトップGIS

どのようなものか

QGISは、地図データを表示、編集、分析するデスクトップアプリケーションです。2002年5月にQuantum GISという名前で始まり、当初の目標は空間データを見るためのビューアーという控えめなものでした。その後、完全な地理情報システムへと育ちました。ファイル、空間データベース、ウェブ地図サービスからレイヤーを開き、1枚の地図に重ね、形状や属性を編集し、分析を行い、結果を印刷地図、地図帳、レポートとしてレイアウトできます。

Open Source Geospatial Foundationの公式プロジェクトで、GNU General Public Licenseの下で公開されています。Qtツールキットを使いC++で書かれており、大半のUnixプラットフォーム、Windows、macOSで動きます。新しいリリースはおよそ4か月ごとです。2.8から始まった方針では、3回に1回を長期サポートリリースとし、次の長期版に置き換わるまで保守します。一方、最新版には毎月バグ修正のポイントリリースが出ます。執筆時点の両者は3.44.15と4.2.3です。

QGISが多くのデータを読めるのは、ほかのプロジェクトの力も使っているからです。ベクトルファイルはGDALのドライバーを通して読み込むため、対応形式の一覧が長く、ラスターデータも同じ仕組みで読み込みます。PostGIS、SpatiaLite、SQL Server、Oracleのテーブルは、元のデータベースから直接読み込みます。公式の機能一覧も、内部形式へ変換することなく、データ本来の形式と投影法で表示すると強調しています。

チームでQGISを共有するようになると、さらに4つの要素が重要になります。

作業の単位はプロジェクトです。`.qgs` ファイルはXMLで、各レイヤーのデータソースと、スタイル、ラベル、フォーム、印刷レイアウトを列挙します。`.qgz` ファイルは、そのXMLと補助データの `.qgd` ファイルをまとめたzipアーカイブです。プロジェクトをGeoPackageやPostgreSQLデータベースの中に保存し、そのプロジェクトが参照するレイヤーと並べて置くこともできます。

Processingは分析の枠組みです。QGIS自身のプロバイダーとGDALのアルゴリズムをまとめたツールボックスがあり、さらにGRASSもプロバイダーとして利用できます。アルゴリズムをグラフにつなぐモデルデザイナー、バッチ処理用の画面、すべての実行履歴も備えます。同じアルゴリズムは、コマンドラインツール `qgis_process` を通じて、デスクトップ画面なしでも実行できます。

```
qgis_process run native:reprojectlayer -- \
  INPUT=roads.shp TARGET_CRS=EPSG:25832 OUTPUT=roads.gpkg
```

PyQGISは、プラグイン、組み込みコンソール、独立したスクリプトを支えるPythonインターフェースです。QGIS Serverは、プロジェクトファイルをウェブで標準OGCサービスとして公開します。WMS、WFS、WCS、OGC API Featuresを提供し、ウェブサーバーの背後でFastCGIまたはCGIアプリケーションとして動きます。

概念を判断に生かす

「QGISとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 船舶、車両、区画、センサーなど、場所に関わる情報を、検索・計測・共有できる地図にまとめます。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 地理情報パイプラインを50倍に高速化 — 50x 地理データパイプラインの高速化

地理情報パイプラインを50倍に高速化 お問い合わせ →

なぜ重要なのか

どれほど優れたパイプラインがあっても、手作業が残る空間データの仕事があります。区画境界と下に敷いた航空写真のどちらが間違っているかを判断する、LiDARの面データから輪郭をなぞる、顧客から届いたデータを取り込む前に点検する、といった作業です。QGISが関わる選択は、その判断をどの道具で行い、判断後に何を残すかです。

前半の選択は、ライセンスと対応プラットフォームで決まります。利用者ごとのライセンスを買う必要はなく、同じリリースをLinux、macOS、Windowsへインストールできます。したがって、外部の受託者、顧客自身の担当者、ビルドサーバーが、プロジェクトを作ったのと同じバージョンを使えます。Esriの商用デスクトップ製品ArcGIS Proは、Windowsでしか動きません。

後半は、作業を終えた後に何が残るかです。QGISはPostGISを直接読むため、編集結果はアプリケーションが配信元に使うテーブルへ入り、後で突き合わせるエクスポートはありません。Processingの履歴にはアルゴリズムとパラメーターが記録され、どの項目もPythonコマンドか `qgis_process` コマンドとして取り出せます。ダイアログで手作業で見つけた修正方法を、それ以降の納品ごとに動くスクリプトの1行にできます。さらにプロジェクトファイルはQGIS Serverからそのまま配信できるので、デスクトップで設定したスタイルが、WMS経由でウェブクライアントの受け取るスタイルになります。

代償の大半は、運用上の規律に関わります。プロジェクトは、自身には含まないデータを参照するため、持ち運べるかどうかは保存したパス次第です。長期版と最新版のどちらを選ぶかにも実質的な違いがあります。前者の変化は緩やかで、後者には4か月ごとに機能が加わります。

実際の活用例

このサイトの事例では、大量の処理を担う道具の傍らで、空間データを目で確かめ、手作業で直す場所としてQGISが登場します。

2015年から2022年までのGISソフトウェア開発チームリーダーの職務では、30件を超えるGISプロジェクトを最初から最後まで担当し、構築、デプロイ、稼働後のサポートまで行いました。その技術構成では、QGISとJOSMが手作業によるデータ処理の場でした。一方、PostgreSQLとPostGISがデータを保持し、GDAL/OGRがshapefile、GeoJSON、GeoTIFF、KML、ベクトルタイルの間でデータを移し、Mapbox GLとLeafletのウェブ地図が表示を担いました。対象は2D・3Dのデジタル地図作成から、LiDAR処理、幾何補正、ベクトル化を経て、屋内地図やナビゲーションまで広がっていました。

2022年から2024年にかけては、大規模な地理空間分析プロジェクトのデータエンジニアとして、地理データと時系列データの1,000件を超える問題を見つけて直す仕事を担いました。GDAL、QGIS、ArcGISを組み合わせて空間的な誤りを診断・修正し、繰り返し必要になるデータのクリーニングはBashスクリプトで自動化しました。空間クエリとインデックスはPostGISが担い、この作業は空間クエリの実行時間を35%短縮することに貢献しました。

問題が起きるところ

デスクトップ画面では空間データが正しく見えてしまい、問題の多くはそこに隠れています。

レイヤーが違う場所に表示される。 QGISは各レイヤーをプロジェクトの座標参照系へその場で再投影します。そのため、各レイヤーが自身の座標系を正しく宣言していれば、異なる系のレイヤーも正しく重なります。`.prj` ファイルのないshapefileは座標系を宣言していません。「Options」の設定によって、QGISは不明のままにするか、座標系を尋ねるか、プロジェクトの座標系または既定の座標系を黙って適用します。対処は、どちらの誤りかで変わります。座標は正しく、指定だけが間違っているなら、「Assign projection」や「Define projection」で指定を変更し、各頂点の位置はそのままにします。座標が実際に別の座標系に属しているなら、「Reproject layer」で変換します。同じツールボックスのグループにある「Find projection」は、レイヤーの範囲が選んだ地域に位置する座標系を列挙し、当て推量を候補の短いリストに変えます。

面積が平方度になる。 式言語には、同じように使えそうに見える2つの面積関数があります。`$area` はプロジェクトの楕円体と単位の設定に従います。楕円体を設定していれば、選んだ単位で楕円体上の面積を返します。`area($geometry)` は常にレイヤー自身の座標系での平面上の計算で、その座標系の単位を返します。そのため、経緯度で保存したレイヤーでは平方度となり、小さい小数値が表の中ではもっともらしく見えてしまいます。人が読む数値には `$area` を使うか、測る前に、その地域に適した投影座標系へレイヤーを再投影します。

プロジェクトを開くとレイヤーの半分が消えている。 プロジェクトが保存するのは各レイヤーのデータの場所であって、データそのものではありません。フォルダーを移す、ファイル名を変える、同僚のマシンで開くといった操作で、保存したパスが参照先を見つけられなくなります。そのときQGISは「Handle Unavailable Layers」ダイアログを表示します。レイヤーを残す選択をすると、プロジェクトは開き、そのレイヤーは一覧に残りますが、各データソースを修復するまで描画されません。プロジェクトのプロパティでパスをプロジェクトファイルからの相対指定にすると、プロジェクトとデータを入れたフォルダーを一緒に移動できます。共同作業では、レイヤーと同じPostgreSQLデータベースやGeoPackageへプロジェクトも保存すると、両者をまとめておけます。

翌月も同じ修正をクリック操作で繰り返す。 ダイアログで直すと、データは正しくなっても、方法は誰かの記憶に残るだけです。次の納品データに同じ問題があれば、また手作業で、少し違うやり方で直すことになります。最初にうまくいった時点でProcessingの履歴から実行内容を `qgis_process` コマンドとして取り出し、納品データを読み込むスクリプトへ入れます。

参考資料

- はじめに — QGISの定義、始まり、公開ライセンスについてのプロジェクト自身の説明です。
- ロードマップ — リリースサイクル、長期版の選び方、現在の予定です。
- 投影法の扱い — 即時の再投影と、座標系のないレイヤーをQGISがどう扱うかを説明します。
- プロジェクトファイルの扱い — `.qgs` と `.qgz` の形式、データベースに保存するプロジェクト、切れたパスの修復を説明します。
- コマンドラインからの処理の実行 — `qgis_process` と、そのコマンドおよびパラメーターの渡し方です。
- 概要 — プロジェクトから公開するサービスについてのQGIS Serverマニュアルの説明です。

用語集

実際の制約に合わせて選ぶ

業務の流れ、既存のインターフェース、変更を必要とする利用者をお知らせください。検証可能な納品内容と、お客様のチームが保守を続けられる引き継ぎ内容を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「QGISとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/qgis/