Digital Corridor Alignment Strategic Framework Plan

Technology Review

June 30, 2023

Task Work Order #28

Contract C9x97

Prepared for:

FDOT Surveying & Mapping Office

1. EXECUTIVE SUMMARY

The Florida Department of Transportation (FDOT) is a leader in surveying and mapping efforts activities statewide. FDOT's Central Surveying and Mapping Office (CSMO) develops policies, procedures, guidelines, and training, in addition to providing overall quality assurance support to the department. While the CSMO leads FDOT's statewide surveying and mapping program, District Surveyors and their staff perform and manage surveying and mapping production efforts supporting Florida's extensive Transportation Work Program, as they collect and maintain business critical horizontal and vertical (i.e., geospatial) survey information.

The survey alignment is used throughout the life cycle of all transportation corridor construction projects and provides the geospatial foundation for all corridor transportation project activities, including the design alignment, the construction alignment, and the Right of Way Map. Surveying of the corridor alignment is the responsibility of the District Surveyor and is second only to the establishment of primary geodetic control in the chronological order of field activities on a transportation project. The data generated by an alignment survey is utilized throughout the life of a construction project. For both 2D and 3D transportation project design, the survey alignment data provides the required geospatial framework. During the right-of-way phase of a project, it is the alignment survey that supports the establishment of legal boundaries of FDOT's right-of-way. During the construction phase, an alignment survey serves to ensure that the as-built construction correctly implements the original design, and that the constructed roadway adheres to the boundaries of FDOT's legal right-of-way in practice.

Often the survey alignment lies within the traveled roadway, requiring surveyors to be in harm's way while recovering physical evidence of the previous survey alignment. As the transportation industry moves away from paper plans, the advantage of a digital survey alignment that could be measured once and used many times is increasing, providing value by reducing project time, project cost, and potentially project lives lost. Finally, a digital survey alignment is the essential geospatial model used in Building Information Modeling (BIM) for Infrastructure that connects all other corridor models both within and between transportation projects. This component is foundational to building and maintaining Florida's smart transportation infrastructure network that may one day support fully autonomous vehicle traffic.

To this end, CSMO leadership has identified the creation of a cloud-based repository of Digital Corridor Alignment (DCA) data as a critical long-term strategic goal that will reduce the cost of doing business and improve worker safety across the state. Three key technical requirements identified by CSMO were:

1. The digital repository and supporting workflows must be able to store, manage, and disseminate geospatial alignment data using the open-source Industry Foundation Class (IFC) 4x3 data model. The IFC model is used by professionals worldwide to manage transportation data. Recently, the American Association of State Highway and Transportation Officials (AASHTO) Board of Directors mandated the adoption of the IFC specification as a basis for the design and construction of roads and bridges by all 50-member State DOT's.
2. The digital repository must be capable of managing, projecting, and transforming geospatial coordinate systems within the database platform, without requiring the use of additional external software.

3. The digital repository and supporting workflows must be able to store, manage, and disseminate geospatial alignment data using database technologies that do not introduce undesired geospatial errors or otherwise reduce the precision and accuracy of the original project survey alignment.

To further this goal, CSMO leadership initiated the development of this project and identified three (3) major tasks to be accomplished:

1. The first task involves a complete technical review of the capabilities of three (3) or more major Relational Database Management Software (RDBMS) platforms at storing, managing, and analyzing geospatial data types.
2. The second task is to complete a "closed loop test" of one or more of the technologies reviewed during the first task to prove the technical viability of storing IFC data in a geospatial RDBMS. During this task, Feature Manipulation Engine (FME) will be used to import a valid IFC 4x3 file and store it in one or more of the RDBMS platforms identified during the technical review and evaluation.
3. The third task utilizes FME to extract a new copy of the IFC data model from the stored copy and make comparisons to the original file to gauge whether FME and the RDBMS technology together can preserve the accuracy and precision of the original alignment data to the fidelity that is required for CSMO's business practices.

The remainder of this report documents the specific activities, results, and lessons learned during the development of this project and is organized as follows:

- **Section 2: Technology Assessment Overview** provides a broad overview of the various technical criteria used by the assessment team that were used to assess the capabilities of five (5) at storing, managing, manipulating, and analyzing spatial data types.
- **Section 3: Technology Assessments** applies the criteria developed in Section 2 to a detailed technical evaluation of each of the five RDBMS technologies selected by CSMO leadership for review. These technologies were: PostgreSQL/PostGIS; Oracle Spatial and Graph; Microsoft SQL Server; Snowflake; and Neo4j.

1.1 Summary of Technology Assessment Findings

A graphical summary comparing the geospatial functionality of the five (5) RDBMS platforms assessed for this report can be found in **Table 1: Technology Assessment Matrix** on page 15.

Out of the five database technologies that were assessed, only PostgreSQL/PostGIS and Oracle Spatial & Graph possess the functionality for managing a wide variety of geospatial data types and formats that will be required to meet CSMO's long-term strategic goals. Both PostGIS and Oracle Spatial are remarkably similar in terms of their overall technical capabilities and geospatial functionality when working with a wide variety of geospatial data models.

There are, however, significant differences in the licensing model and overall cost between the two platforms. CSMO leadership has expressed a strong preference for software that is licensed under an open-source license, as well as their concerns about both the long-term cost of Oracle's commercial licenses.

The other three technology platforms assessed offer rather anemic and lackluster support for spatial data. Based on an assessment of their technical capabilities alone, none of the other three RDBMS platforms that were investigated during the technology review process will be an appropriate platform capable of meeting CSMO's business goals and needs for DCA data management.

Table of Contents

Figures

Tables

Glossary, Acronyms, and Abbreviations

CSMO. FDOT Central Surveying and Mapping Office

DCA. Digital Corridor Alignment data.

ELA. Enterprise License Agreement.

ETL. Extract, Transform, and Load.

FDOT. Florida Department of Transportation.

FME. Commercial Extract-Transform-Load software platform developed and sold by Safe Software. Also known as Feature Manipulation Engine or FME Safe.

GeoJSON. Geographic JavaScript Object Notation. An extension of the popular text-based JSON data format capable of encoding geospatial data types.

GDAL. Geographic Data Abstraction Library, a major open-source GIS project that maintains a variety of powerful, command line tools for managing spatial data.

GIS. Geographic Information Systems.

GML. Geographic Markup Language. A text-based exchange format for geographic data.

IFC. Industry Foundation Class.

IT. Information Technology.

KML. Keyhole Markup Language. Geographic data format popularized by Google Earth.

Lidar. Light Detection and Ranging. Lidar datasets are a key dataset for digital twin technologies.

LRS. Linear Reference System.

NSRS. National Spatial Reference System. The NSRS is maintained by the National Geodetic Survey (NGS) as the standard coordinate system that is used by all surveyors in the US. The NGS is expected to publish a major update to the NSRS in the next 6-24 months.

NGS. National Geodetic Survey. The NGS is the federal agency that maintains the standard coordinate and projection systems and mathematics used by all surveyors within the US.

NOAA. National Oceanic and Atmospheric Organization. NOAA is the parent agency of the National Geodetic Survey (NGS).

OIT. FDOT's Office of Information Technology

OGC. Open Geospatial Consortium

OGC SFA. Open Geospatial Consortium (OGC) Simple Feature Access (SFA). Also known as *OpenGIS® Implementation Standard for Geographic Information - Simple feature access - Part 1: Common*

*Architecture (v1.2.1)***.** OGC SFa IS one of two industry standards for implementing spatial vector features in a SQL-compliant database.

OSGeo. Open-Source Geospatial Foundation

proj. Open-Source project for coordinate system projection and transformation

RDBMS. Relational Database Management Software/System. An RDBMS is the 'traditional' technology for storing and retrieving enterprise data and in the context of this report is synonymous with 'enterprise database platform.'

RDF Triples. Resource Description Framework triple. A graph data structure that details subject-predicted-object relationships among entities. Directed graph data structures are conceptually quite different from data structures based on the more traditional related table data models used by a traditional RDBMS. Synonymous with **Semantic Triples.**

ROW. Right-of-Way.

Semantic Triples. See **RDF Triples.**

SPCS. State Plane Coordinate System. The SPCS is the federal standard for producing maps that require a high degree of accuracy with regards to real-world locations.

SQL. Structured Query Language. SQL is the standard query language for most relational database management systems.

SQL/MM. The SQL Multimedia Standards document, also known as the ISO/IEC 1349-3 SQL Multimedia Spatial Standard**.** The SQL/MM standard extends the **OGC SFA** standard to include curve geometries. Together, these two technical standards documents describe the vector data types that are implemented into most spatial databases that utilize the SQL language.

[This page intentionally left blank]

2. TECHNOLOGY ASSESSMENT OVERVIEW

In the field of geospatial professionals, managing and analyzing a wide variety of specialized data types is an essential requirement for any RDBMS. In coordination with CSMO senior leadership, the review team developed multiple technical review criteria identified as mission-critical items that would be essential for supporting CSMO's forward-looking goal to build and maintain a central data repository for managing DCA datasets collected statewide.

The review criteria developed by the review team are:

- A review of the **long-term history** of each technology, from the date of original release to the most recent version. Long-term projects that are still in active development and are widely used by a large number of enterprise users and organizations are naturally preferred over smaller and more obscure technologies.
- The **licensing model and cost** for each technology. CSMO has expressed a desire to remain within the open-source ecosystem, if possible.
- Support for a wide variety of **vector data types**, particularly the geography and 3D/4D subtypes. Both of CSMO's DCA datasets and the IFC data model are vector-based data that represent points, lines, polygons, curves, and surfaces.
- **Coordinate system management and the ability to reproject and transform geospatial coordinates** in one coordinate system to another coordinate system without invoking 3rd party or external software. The ability to manipulate coordinate systems and projections natively, without invoking additional external software to do so, was identified by CSMO senior technical personnel as a critical technical requirement due to ongoing changes to federal survey and data standards. The National Oceanic and Atmospheric Administration's (NOAA) National Geodetic Survey (NGS) is the US federal agency that defines and manages the National Spatial Reference System (NSRS) and the State Plane Coordinate System (SPCS). The NSRS is the national framework and foundational datum for all geospatial positioning measurements within the US and its holdings worldwide. The NGS is preparing to release an updated NSRS in 2025. The impending release of the updated NSRS will have a more accurate earth origin allowing for precise measurements of the Earth's tectonic plate movements. This will require existing survey measurements to be transformed at certain epochs to maintain their surveyed position relative to the NSRS as it physically exists over time.
- **Compatibility with FME**, including the ability to successfully ingest, transform, and store a wide variety of spatial data types. This is a critical technical requirement for building digital twin data models that seek to replicate physical, real-world objects to a high degree of detail and fidelity.
- **Raster data support**, for storing and managing raster datasets. To key raster data types are aerial imagery and Digital Elevation Models (DEM) derived from extremely high-resolution lidar point cloud datasets. While not strictly required for Digital Corridor Alignment (DCA) datasets, both high-resolution aerial imagery and high-resolution lidar DEMs are key 'value-add' datasets that, when combined with DCA datasets, provide a technical basis for building 'digital twins' of real-world objects.

- **Lidar Point Cloud support**, for storing and managing both terrestrial and aerial Light Detection and Ranging (Lidar) datasets. Lidar point clouds are extremely dense sets of laser distance measurements taken from a highly specialized lidar collection device and are a key technology driving the digital twin concept. These devices are essentially a large-format laser scanner that allow real-world objects to be very precisely measured and modeled in digital form. Broadly, lidar data is collected in one of two ways: aerial lidar is collected from a fixed-wing aircraft and images objects from an orthometric (top-down) perspective, while terrestrial lidar is collected from ground-based platforms and provides an oblique (side-view) perspective of physical objects. Taken together, aerial and terrestrial lidar allow for the creation of extremely precise and accurate digital models of as-built real-world objects and assets, such as pavement, drainage, bridges and buildings.
- Support in **Microsoft Azure** and **Official OIT support** status. These criteria are key organizational requirements that will ensure the long-term support and health of any digital platform adopted by CSMO.
- Support for **Resource Description Framework (RDF) triples and directed-graph data structures**. The object-oriented structure of the IFC data model describes the abstract relationships between the various entities that make up an alignment dataset.

The remainder of **Section 2** provides a brief explanation and a limited technical background for each of the spatial data types that were defined as assessment criteria, while **Section 3** contains the individual technical assessments for each of the five database platforms.

2.1 Vector Data

Vector data is a spatial data type that represents real-world objects as sets of discrete geometric shapes, with these, typically, being points, lines, and polygons. FDOT uses the vector data type to record, store, and map a wide array of mission-critical data, such as the locations of signaling devices (points), roadways (lines), and various administrative boundaries (polygons). Support for vector data is a critical technical requirement for CSMO's DCA data, as support for 3D data types is needed. This is due to the data being recorded in the DGN or DWG file formats, both of which are vector file formats utilized by commercial CAD software.

Geometries vs Geographies

There are two general types of vector data types: geometries and geographies. The major difference between the two lies in the type of underlying coordinate system that is used to represent real world locations.

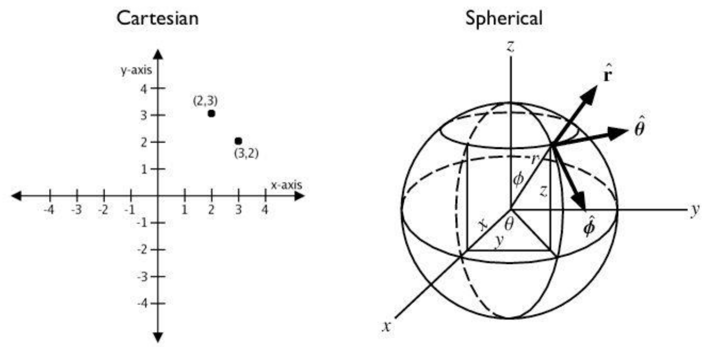

FIGURE 1: CARTESIAN COORDINATE SYSTEM (GEOMETRY) VERSUS A GEODETIC COORDINATE SYSTEM (GEOGRAPHY)

FIGURE 2: STANDARD VECTOR TYPES DEFINED BY THE OGC SFA AND SQL/MM STANDARDSFIGURE 3: CARTESIAN COORDINATE SYSTEM (GEOMETRY) VERSUS A GEODETIC COORDINATE SYSTEM (GEOGRAPHY)

A geometry, such as a DCA, stores points, lines, curves, and polygons using a Projected Coordinate System, which represents geographic features using X and Y values on a flat plane. Calculations, like distance and area, tend to be easier and faster when using a geometry data type instead of performing comparable calculations using the geography data type. The ease and speed of spatial calculations when using geometries is balanced by the fact that all geometries inherently introduce some level of location errors into spatial data. Locations on the surface of a sphere or an ellipsoid (i.e., real-world locations on the earth's surface) cannot be mathematically transformed into locations on a flat plane without introducing various discontinuities and distortions.

In contrast, the geography data type stores points, lines, curves, and polygons using geodetic coordinate systems, which represent locations as latitude and longitude values that map to a spherical or ellipsoidal model surface. Geographies have the advantage of introducing little additional error due to how they store real-world locations. Here, an underlying surface that is more "true to life" to the Earth's surface is used over the flat plane used by the geometry data type.

The major drawback to the additional accuracy and real-world fidelity offered by the geography data type is a significant increase in the difficulty of spatial calculations and a large performance penalty for even simple calculations. Calculations, like distance and area, in a geodetic coordinate system require the use of spherical trigonometry, rather than the more familiar and simpler methods from cartesian geometry. As a result, the geography data type inevitably has less functionality, as implementing even simple calculations tends to be much more difficult in practice. Moreover, when a spatial database does implement spatial calculations for geographies, the same calculations performed against the geography data type can be orders of magnitude slower than the same calculation executed against a geometry. This is due to the greater level of computational complexity associated with using a geodetic coordinate system.

Standard Vector Object Types

There are two standards that define the standard vector object types that can be stored in SQL-compliant databases.

The vector object types described by the Open Geospatial Consortium, Inc. in its *OpenGIS® Implementation Standard for Geographic Information - Simple feature access - Part 1: Common Architecture (v1.2.1)* are reference vector data types that are used across the GIS and database industries.[1] This standard is usually referenced as the Open Geospatial Consortium Simple Feature Access (OGC SFA) standard, and it provides a complete taxonomy of simple vector types as well as their associated properties independently of any particular software environment or system. OGC SFA is designed to be an accessory standard that is compatible with and expands the wider SQL language standards adhered to by most RDBMS software.

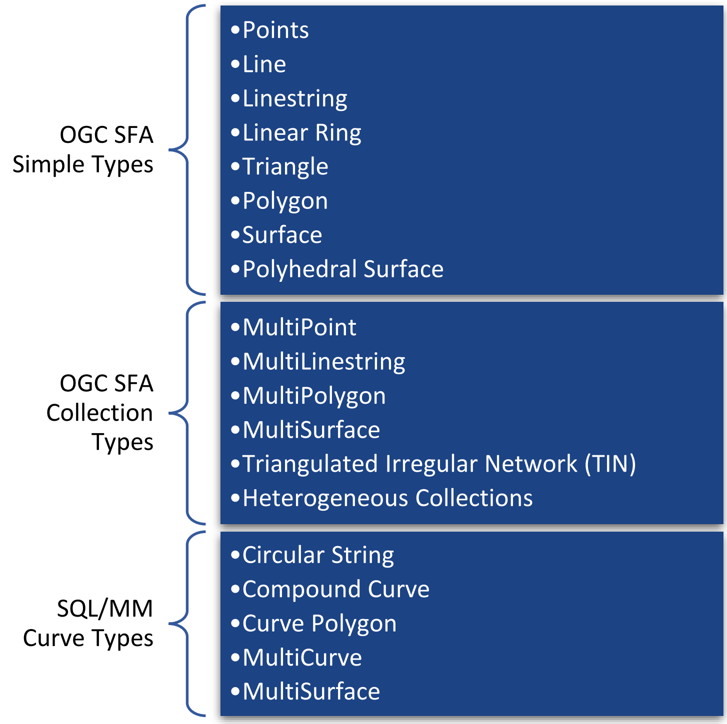

FIGURE 4: STANDARD VECTOR TYPES DEFINED BY THE OGC SFA AND SQL/MM STANDARDS

The ISO/IEC 1349-3 SQL Multimedia Spatial Standard, commonly referred to as the SQL/MM standard, extends the OGC SFA standard vector data types to include geometry subtypes that are composed of curves with circular arcs rather than line strings.[2] These vector data types are less frequently used, but in certain situations, they can allow for a more trusted and accurate representation of real-world features than the OGC SFA geometry types alone.

Taken together, the OGC SFA and SQL/MM standards define the universe of standard vector objects that can be stored in a SQL-compliant RDBMS. In practice, each RDBMS software environment implements a subset of the vector types described by these two standards. The amount and types of vector data that are supported by any particular RDBMS serve as important indicators of its performance in storing, querying, and analyzing spatial data.

Geometries

Geometries are two-dimensional (2D) representations of geographic objects located on a flat plane. The shortest path between two points on a plane is a straight line, and the spatial properties of geometries (areas, lengths, distances, intersections, etc.) can be quickly and easily calculated using straight line vectors and basic cartesian mathematics.

---

[1] "Simple Feature Access," 2023

[2] "ISO/IEC," 2016

This ease of computation is a tradeoff against the fact that storing spatial data as a geometry automatically introduces some level of error into real world locations and measurements. All geometries use *Projected Coordinate Systems* that model the Earth's surface as a flat surface, and no single projected coordinate system can accurately represent the spherical surface of the earth. Geometries also accumulate real-world error when they undergo multiple transformations and projections between various coordinate systems. They can also produce incorrect and inconsistent results if the underlying projected coordinate system is not appropriate for the underlying scale of the data or the area on the Earth's surface.

Geographies

Geographies represent point, line, and polygon features on a spherical surface. Geographies use angular coordinate systems that are measured from the center (i.e., geographic latitude and longitude) rather than X and Y locations on a flat plane. The geography data type is particularly useful for applications that require calculations based on the curvature of the Earth's surface, such as distance and area measurements over large geographic areas. This data type is associated with a high degree of real-world location accuracy.

Calculations using the geography data type are frequently much more complex than the corresponding calculation for the geometry data type. While the shortest path between two points on a geometric plane is a straight line, the shortest path on the surface of a sphere is a great circle arc. Calculations using the geography data type utilize spherical trigonometry to compute even the most basic of measurements. For this reason, calculations on a spherical surface tend to be more difficult to implement and computationally expensive in comparison to the same calculations using the geometry data type. In general, the geography data type will typically have less functionality in a spatial database than does the simpler geometry data type, and the same calculations will require more time to process.

The generally reduced functionality and increased computational difficulty of the geography data type is a trade-off against the higher level of accuracy and trusted representation of real-world location data that geographies provide. Despite their reduced functionality and slower computation speed, the geography data type is ideal for storing data that covers large geographic areas, requires a high degree of real-world accuracy, or both.

3D and 4D Vector Data

Three- and four-dimensional (3D, 4D) data types, 3DM, 3DZ, and 4DZM geometries and geographies, store additional values along with the planar or geodetic X and Y locations. In the standard language of spatial databases, M represents a measure associated with a location such as the measure of the distance of a vertex from the origin of the line, while Z typically represents an elevation value in a 3D space.

Vector features with M values are called 3DM vectors. These vector types are typically used for linear referencing features against a network of lines. Linear referenced coordinates are vital to many FDOT business practices, allowing for roadway segments to be tied into the department's Roadway Characteristics Inventory (RCI) database maintained by the Transportation Data and Analytics (TDA) Office.

Vector features with Z values are called 3DZ geometries. These geometry types store the location of features in 3D space, with the Z value representing an elevation value. Measurement of accurate elevation values on the earth's surface is a technical topic that is fraught with complexities, but in most cases, elevations are referenced to one of two models of the earth's surface within a spatial database: an ellipsoidal model or a geoidal model. An ellipsoidal model uses an idealized flattened sphere for its model of the earth's surface, while a geoid is an irregular model of the earth's surface that is based on mapping variations in the strength of gravity.

A four-dimensional vector feature (4DZM) stores values for both a measure and an elevation.

Topologies

Topologies are a special case of vector data that are particularly useful in a variety of applications, including transportation and utility networks. Topologies store the relationships between and connectivity among geographic features in addition to the geographic features themselves. The connections and relationships among geographic features can be a critical aspect to managing geospatial data, particularly transportation data.[3]

2.2 Coordinate Systems and Transformations

Coordinate system management and coordinate transformations and projections are essential for accurately representing geospatial data. The ability to define and manipulate coordinate systems ensures that data from multiple sources, often recorded in different coordinate systems depending on their locations, can be accurately represented, integrated, and analyzed together in a single database.

CSMO has identified the ability to define coordinate systems and transform them using industry standard transformations as a critical requirement for any future DCA digital archive.

2.3 FME and GDAL Compatibility

When dealing with large amounts of data in differing formats, compatibility with FME and GDAL is critical. FME empowers users in the movement, transformation, and integration of data across different systems and formats. Similarly, GDAL helps users with a variety of ETL and management tasks, aiding in the conversion of geospatial data between different formats. These technologies provide users with enhanced capabilities in data management, transformation, and more. This functionality helps to increase the interoperability among multiple, disparate sources of data and is a critical requirement for organizations that are seeking to synthesize and mine multiple types of data for actionable business insights.

FME Compatibility

Safe Software's FME is a data integration platform that enables users to move, transform, and integrate data between various systems and formats. FME was developed in response to the increased demand for data integration and transformation tools designed to be implemented universally. Existing tools were often designed for specific use cases and had limited functionality when applied to more general ETL tasks.

FME provides a graphical user interface for designing workflows or workspaces that specify how data should be transformed or converted. Workspaces can be saved and re-used, making it easy to automate data integration tasks. In FME, readers, writers, and transformers are components used to process data as it moves through a data transformation workflow. Readers are components used to read data from a source format, while writers are used to write data to a destination format. Similarly, transformers are

[3] Typically, the terms 'node,' 'edge,' and 'face' are used when discussing topological point, line, and polygon features to distinguish topology-aware vector features from geometries and geographies. This convention is maintained in this section.

components used to manipulate, convert, or validate data as it moves through the workflow. Transformers can perform a wide range of tasks, including spatial analysis, data validation, attribute manipulation, and more.

The use of a graphical user interface is beneficial for visualizing data processes, but there is an inherent learning curve for inexperienced users to become familiar with the hundreds of different transformers offered. There is a highly active user community that provides feedback and troubleshooting advice. FME supports over 450 data formats and systems, including spatial data formats such as ESRI Shapefiles, GeoJSON, KML, and PostGIS. It also supports non-spatial formats such as CSV, Excel, XML, and JSON, as well as databases like Oracle, SQL Server, MySQL, and PostgreSQL. One of the key features of FME is its ability to perform complex data transformations, including spatial data transformations, making it a valuable tool for GIS professionals. FME provides an extensive library of transformers, and these pre-built tools can perform specific data transformations. Users can also develop their own transformers using FME's Python or C++ APIs.

FME can be run in a desktop, server, or cloud-based environment. The FME Server component allows workspaces to be published as web services, making it easy to share and distribute data integration tasks across an organization or with external partners. FME is proprietary commercial software and requires a paid license to use.

GDAL Compatibility

The Geospatial Data Abstraction Library (GDAL) is a popular open-source library for reading, writing, and processing raster and vector geospatial data formats. GDAL is widely used in the GIS community for ETL and management tasks, and it provides a set of command-line utilities for converting between different geospatial data formats. GDAL also allows for users to perform a variety of geospatial management and analytic tasks, including the reprojection of data between different coordinate systems. GDAL provides a comprehensive application programming interface (API) and multiple language bindings for developers to access its functionality from their own software applications.

GDAL is already incorporated in many GIS software environments, including both ESRI’s commercial ArcGIS Pro and the open-source QGIS Desktop, both of which provide a graphical ‘skin’ to access GDAL’s command line tools via their own graphical user interfaces.

2.4 Azure RDS and OIT Support

Azure

Azure RDS (Relational Database Service) is a fully managed database service provided by Microsoft Azure. It allows users to easily create and manage cloud-based relational databases without needing to worry about the underlying infrastructure or database administration tasks. Azure RDS supports several popular relational database engines, including Microsoft SQL Server, MySQL, and PostgreSQL. It provides features such as automated backups, high availability, and automatic scaling of computing resources based on usage patterns.

Users can provision an Azure RDS instance by selecting the desired database engine, configuring the computing and storage resources, and specifying other settings such as the database name and administrative credentials. Once provisioned, users can connect to the database instance using standard database connectivity protocols, and perform standard database operations such as creating tables, inserting data, and running queries.

One of the key benefits of Azure RDS is its scalability. Users can easily scale up or down the computing and storage resources of their database instance as needed, without needing to worry about downtime or data migration. Azure RDS also supports automatic scaling, which allows the service to automatically adjust the computing resources based on usage patterns, ensuring optimal performance and cost efficiency. Another important feature of Azure RDS is its security capabilities. Azure RDS supports network isolation, data encryption, and access control features such as Azure Active Directory integration and firewall rules, providing a secure environment for storing and processing sensitive data.

FDOT Office of Information Technology Support

The FDOT Office of Information Technology (OIT) plays a critical role in the management of the Department's information technology systems and infrastructure. OIT is responsible for creating and maintaining the enterprise technology standards used by the Department, as well as the software and hardware infrastructure that supports the FDOT's mission. Due to OIT's important role within the Department, ensuring that technological solutions adopted by CSMO fit within the enterprise standards and requirements outlined by OIT will facilitate the long-term success of CSMO's centralized DCA repository.

2.5 Raster Data Support

Raster data is a type of geospatial data for representing images and continuous surfaces. For example, digital elevation models and aerial images are commonly stored in raster data formats. Raster's are made up of a grid of cells with a value assigned to each cell. The value within the cell depends on the information being represented and can be single band data (i.e., elevations) or multiband data (i.e., RGB values in an aerial image).

While not immediately critical for archiving DCA data, the ability to combine DCA data with high-resolution digital elevation models and aerial imagery is an extremely compelling "value-add" that will allow CSMO to leverage additional value out of the original corridor alignment survey data.

2.6 Lidar Point Cloud Support

Lidar (Light Detection and Ranging) point clouds are a type of geospatial data that are collected using lasers mounted on aircraft, drones, vehicles, static equipment, and even handheld systems. The lasers emit pulses of light that bounce off the surface of the earth and return to the sensor, allowing for the creation of a detailed three-dimensional representation of the terrain. Lidar point clouds consist of millions of data points that can be used for a variety of applications, including topographic mapping, land-use planning, and urban development. Lidar point clouds are a valuable tool for gaining insights into the physical landscape and are widely used in the field of geospatial analysis.

2.7 Support for Resource Description Framework Triples and Directed-Graph Data

The Resource Description Framework (RDF) was originally proposed as a recommendation to the World Wide Web Consortium in 1999. The RDF framework was originally conceived as a foundation for processing metadata about entities and their relationships, with a broad goal to "define a mechanism for describing resources that makes no assumption about a particular application domain, nor defines (a priori) the semantic of any application domain."[4] Today, the RDF data structure - commonly referred to as 'RDF triples' or 'semantic triples' – has become the common pattern for encapsulating information about entities and the relationships among them in a directed graph data structure.

An RDF triples utilize the basic design pattern of *Subject -> Predicate -> Object* to denote the nature of the relationship (the predicate) between the subject and object. An example in common language would be a phrase like "Florida's Turnpike Enterprise is a part of FDOT," which maps to the semantic triple *Florida's Turnpike Enterprise → is a part of → FDOT*. This general form is extensible and can include both generic meta information about an IFC entity (*Roadway → is → I-95)* as well as explicit relationships between IFC entities (*Alignment point → is related to → geodetic control point).*

In order to fully leverage the object-oriented nature of the IFC data model over the long-term, the ability for an enterprise database platform to natively support the directed-graph data model and graph-based queries is an important consideration. Database platforms that cannot support directed-graph data models natively would require a separate and additional graph database (e.g. neo4j) in order to fully capture the relationships that are embedded in the IFC data model.

3. Technology Assessments

Five database technologies were selected by CSMO leadership for technical assessment:

1. PostgreSQL/PostGIS
2. Oracle Spatial
3. Microsoft SQL Server
4. Snowflake
5. Neo4j

At the time of writing, these five technologies were among the most popular Relational Database Management Systems (RDBMS) used across the world.[5]

Out of the five database technologies that were assessed, only two - PostgreSQL/PostGIS and Oracle Spatial & Graph - possess the underlying functionality for managing a wide variety of geospatial data types and formats that will be required to meet CSMO's long-term strategic goals.

Both PostGIS and Oracle Spatial are remarkably similar in terms of their overall technical capabilities and geospatial functionality when working with a wide variety of geospatial data types, with Oracle Spatial and Graph being the slightly more technically capable but more expensive solution.

While CSMO leadership has expressed interest in software that is licensed under an open-source license, they remain open to the possibility of using Oracle. Although Oracle could be associated with long-term costs for their commercial licenses, CSMO is aware of other FDOT business units that utilize Oracle for their database needs.

The other three technology platforms that were assessed in the section offer rather anemic support for spatial data types. Based on an assessment of their technical capabilities alone, none of the other three

[4] https://www.w3.org/TR/PR-rdf-syntax/
[5] "DB-Engines Ranking," 2023

RDBMS platforms that were investigated during the technology review process will be an appropriate platform capable of meeting CSMO's business goals and needs for DCA data management.

A summary matrix containing both the assessment criteria and the findings for all five (5) enterprise RDBMS can be found on the following page and more detailed information with supporting citations can be found within each technology subsection.

General Information	PostgreSQL/PostGIS	Oracle Spatial	SQL Server	Snowflake	neo4j
License	PostgreSQL License GNU General Public License	Commercial	Commercial	Commercial	Open-Source Community Edition Commercial Edition
Cost	Free (Both)	Commercial via FDOT ELA	Commercial via FDOT ELA	Priced by Usage	Free (Community Edition) Commercial Edition
Release Year	1986 (PostgreSQL) 2001 (PostGIS)	1979	April 24, 1989	October, 2014	February 2010 (v1.0)
Current Release	PostgreSQL v15.2 (February 2023) PostGIS v3.3.2 (April 2023)	21c (January 2021)	SQL Server 2022 (November 16,2022)	N/A	v5.6 (March 2023)
Paid Enterprise Support	✓	✓	✓	✓	✓
OIT Support	✗	½	✓	½	✗
RDF/Graph Data Model Support	Apache AGE	Oracle Graph	SQL Graph	✗	✓
Geospatial Support					
Vector Data	✓	✓	✓	✓	✗
Geometry	✓	✓	✓	✓	Point Geometries Only
Geometry Functions	250+				3
Geography	✓	✓	✓	✓	Point Geographies Only
Geography Functions	31				3
Vector 3D & 4D	✓	✓	✓	✓	
3DM Support	✓	✓	Read/Write Only	✓	✗
3DZ Support	✓	✓	Read/Write Only	✗	Point Geographies/Geometries Only
4DZM Support	✓	✓	Read/Write Only	✗	✗
Topology & Network Data	✓	✓	✗	✗	✓
Coordinate Systems	✓	✓	✗	✗	✗
Custom Coordinate Systems	✓	✓	✗	✗	✗
Projections/Transformations	✓	✓	✗	✗	✗
Custom	✓	✓	✗	✗	✗
Raster Data	✓	✓	✗	✗	✗
Raster Functions	288	✓	✗	✗	✗
Supported Raster Formats	146 filetypes (via GDAL)	✓	BLOB Only	✗	✗
Lidar Point Cloud	✓	✓	✗ BLOB Only	✗	✗
Extract-Transform-Load					
Native Spatial ETL Tools	✓	✓	✓	✓	✗
GDAL Compatible	✓	✓	✓		✗
FME Compatible	✓	✓	✓	✓	JDBC Only

TABLE 1: TECHNOLOGY ASSESSMENT MATRIX 1

3.1 PostgreSQL/PostGIS

PostgreSQL began as the Defense Advanced Research Projects Agency (DARPA) sponsored POSTGRES project at the University of California at Berkeley in 1986. In 1994, a SQL interpreter was added to the POSTGRES95[6] release, and PostgreSQL took its current form as a SQL-compliant enterprise RDBMS. Today, the PostgreSQL project has a robust community of users, developers, and financial sponsors who contribute time and money to feature development, helping PostgreSQL to become one of the most popular enterprise data storage technologies in the world.

FIGURE 7: THE POSTGRESQL LOGO

PostgreSQL has seen continual growth in adoption as an enterprise database platform since its initial release, due to both the range of technical features it offers as well as its underlying licensing model. PostgreSQL is licensed under the custom PostgreSQL License, a permissive open-source license that is based on the popular MIT license and allows users to utilize, modify, or alter the base PostgreSQL software in any manner that they see fit for any purposes, including commercial. Many large, well-known companies and websites utilize PostgreSQL to power their online offerings and enterprise IT infrastructure, including Apple/OS X, the Internet Movie Database (IMDB), Instagram, Reddit, Skype, Spotify, Twitch, and the International Space Station.[7] Overall, PostgreSQL's low cost and support for a wide variety of data types have made it an appealing option for organizations that manage large volumes of data at enterprise scales.

Spatial data is supported in PostgreSQL via the PostGIS extension. PostGIS is an open-source extension for PostgreSQL that allows spatial data to be stored, managed, and analyzed within a PostgreSQL database. The PostGIS project is a separate project from the main PostgreSQL development project, and it is currently managed by the Open-Source Geospatial Foundation (OSGeo) as a core project.[8] The PostGIS extension is licensed under the GNU General Public License (GPL), a widely used 'copy-left' license used in the open-source community. A GPL license allows users to utilize PostGIS for any purpose, including commercial, but it requires that any changes or modifications to PostGIS's source code be provided to the end-user upon request and free of charge.

PostgreSQL and PostGIS are both supported on Windows, and Microsoft Azure supports PostgreSQL and PostGIS in a variety of deployment modes, including as a managed Azure Relational Database Services (RDS) instance. In recent years, the popularity of PostgreSQL/PostGIS among large enterprise users has reached a tipping point and a new ecosystem of third-party companies offering 24/7 enterprise support for the deployment of PostgreSQL and PostGIS at scale have entered the market.[9,10]

PostgreSQL and PostGIS support all the major spatial data formats and functionality identified in this report as critical to the business practices of the FDOT Surveying and Mapping Office.

[6] "A Brief History of PostgreSQL," 2023
[7] "Companies That Use PostgreSQL," 2023
[8] "PostGIS-OSGeo," 2023
[9] "EDB Enterprise Plan," 2023
[10] "Support – PostGIS," 2023

General Information	PostgreSQL/PostGIS
License	PostgreSQL License GNU General Public License
Cost	Free (Both)
Release Year	1986 (PostgreSQL) 2001 (PostGIS)
Current Release	PostgreSQL v15.2 (February 2023) PostGIS v3.3.2 (April 2023)
Paid Enterprise Support	✓
OIT Support	✗
RDF/Graph Data Model Support	Apache AGE
Geospatial Support	
Vector Data	✓
Geometry	✓
Geometry Functions	250+
Geography	✓
Geography Functions	31
Vector 3D & 4D	✓
3DM Support	✓
3DZ Support	✓
4DZM Support	✓
Topology & Network Data	✓
Coordinate Systems	✓
Custom Coordinate Systems	✓
Projections/Transformations	✓
Custom	✓
Raster Data	✓
Raster Functions	288
Supported Raster Formats	146 filetypes (via GDAL)
Lidar Point Cloud	✓
Extract-Transform-Load	
Native Spatial ETL Tools	✓
GDAL Compatible	✓
FME Compatible	✓

TABLE 1: POSTGRESQL ASSESSMENT MATRIX

Geometry Support

PostGIS natively supports both the standard OGC SFA vector types as well as the SQL/MM extended curve types.[11]

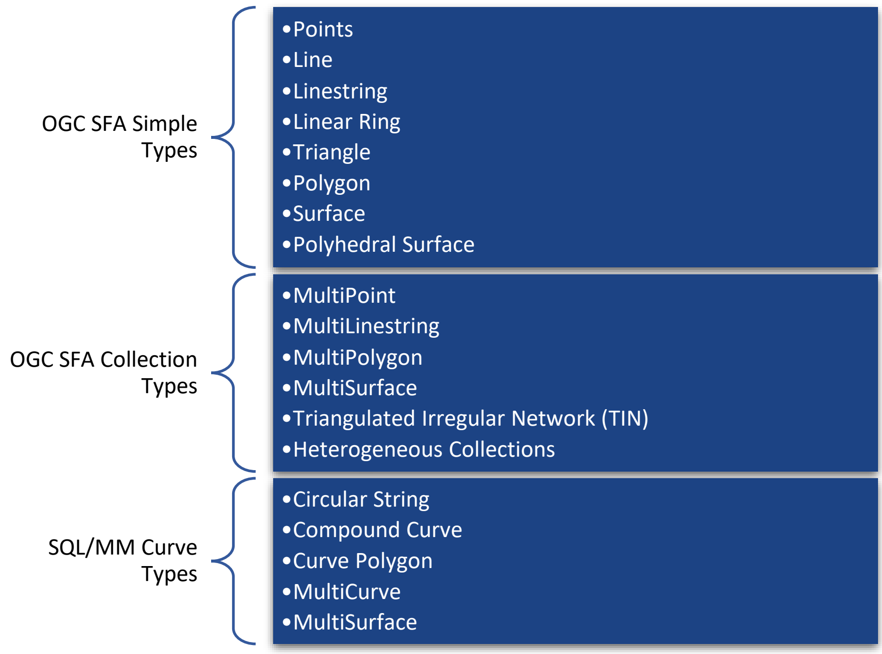

FIGURE 4: POSTGIS VECTOR DATA TYPES

PostGIS has a large number of functions that operate on geometry data types, and a query of the *pg_catalog.pg_proc* table in PostGIS v3.3.2 returned 227 distinct functions that take one or more geometries as an input parameter. These spatial functions include options for geometry construction, access, editing, validation, processing, and clustering. Other common spatial tasks, like measuring distances and areas, also have functionality available. PostGIS also includes additional sets of functions for performing specialized tasks that are commonly used in conjunction with FDOT data, such as linear referencing locations to a linear referencing system (LRS).

A full listing of all 227 functions for working with vector data can be found in the PostGIS reference manual or in the PostGIS Function Support Matrix.[12,13]

Geography Support

A search of *pg._catalog.pg_proc* in PostGIS v3.3.2 returned 25 functions for working with the geography data type as an input parameter. Most basic vector functions are available for geographies. These include measuring length, distances, perimeters, and areas; finding features within a threshold distance of another feature; and determining if two features spatial intersect, cover, or are covered by each other.

The PostGIS manual recommends a standard work-around for cases where a particular spatial function is necessary for a workflow but is not compatible with the geography data type. This alternative is to cast a geography to a geometry using the *ST_Transform()* function and an appropriate Projected Coordinate System, perform the required calculation, and then cast the results back into a geography by calling ST_Transform() again. Doing so will incur a small penalty in terms of performance, but it allows for PostGIS's spatial functionality to be used with the geography data types.

---

[11] "Chapter 4. Data Management"
[12] "Chapter 8. PostGIS Reference"
[13] "Chapter 15. PostGIS Special Functions Index"

3D & 4D Support

PostGIS supports 3DM, 3DZ, and 4DZM data types for all vector object types, for both geographies and geometries. Elevation values are referenced to an ellipsoid surface; PostGIS does not support referencing elevations to a geoid. PostGIS has 127 spatial functions that support using 3DM, 3DZ, or 4DZM vector objects as inputs.

Topology Support

PostGIS supports topological data structures via the *postgis_topology* extension. This extension enables efficient spatial analysis through the enforcement of spatial relationships (topologies) between vector features. PostGIS includes approximately 100 functions for constructing, editing, and analyzing topology data as well as enforcing spatial relationships between vector features.[14]

Support for Coordinate Systems and Transformations

PostGIS offers a comprehensive set of coordinate systems and transformation functions by integrating the open source *PROJ* tool into the PostGIS back-end.[15] At the time of writing, PROJ v6.3.1 is installed by default as a part of PostGIS v3.3.2.4975da8, and it provides the user with the ability to store and convert between 8,501 separate coordinate systems.

PROJ is released under the X/MIT open-source license, a permissive open-source license that does not provide any downstream restrictions on use and is frequently used as a component of both commercial and open-source GIS software.

Extract Transform Load Support

PostgreSQL supports a variety of ETL capabilities through either built-in functionality or third-party tools and extensions. Some ETL capabilities include the following:

- Data Extraction: PostgreSQL supports the ability to extract data from a variety of sources, including other databases, flat files, and web services. The PostgreSQL COPY command can be used to import data from CSV, TSV, and other file formats.[16]
- Data Transformation: PostGIS provides a rich set of functions for transforming geospatial data, including conversion between coordinate systems, buffering, and simplification. PostgreSQL also provides support for user-defined functions, which can be used to perform custom data transformations.
- Data Loading: PostgreSQL supports the ability to load data into a database using either the COPY command or a variety of third-party ETL tools. Additionally, PostgreSQL supports the ability to load data in parallel through the utilization of the built-in parallel query feature.

FME Support

PostgreSQL is compatible with FME. FME supports reading and writing data to/from PostgreSQL databases using its built-in readers and writers. Additionally, FME can also perform complex data transformations, including spatial data transformations, on data stored in PostgreSQL databases. FME also supports

[14] "Chapter 10. Topology"
[15] Evenden et al. "PROJ," 2023
[16] "Copy: PostgreSQL," 2023

PostGIS, and together the two software platforms provide a wide range of spatial data management and analysis functions that can be leveraged by FME to perform advanced spatial data processing.[17]

GDAL Support

PostGIS is tightly integrated with GDAL and is natively available in PostgreSQL by enabling the PostGIS extension. Using GDAL as a backend provides PostGIS the ability to store, query, and analyze a much larger variety of geospatial data types within PostgreSQL databases. Once the PostGIS extension is enabled, GDAL can be used to read and write geospatial data to and from the PostgreSQL database.[18]

Azure RDS

PostgreSQL can be used with Azure RDS. With Azure RDS, you can set up, operate, and scale PostgreSQL databases in the cloud without having to manage the underlying infrastructure.

To use PostgreSQL with Azure RDS, you would need to create a new PostgreSQL database instance in the Azure portal, specifying the desired configuration settings such as the instance size, storage capacity, and backup options. Once the database instance is created, you can connect to it using standard PostgreSQL client tools. Azure RDS also provides various features for managing PostgreSQL databases, such as automatic software patching, automated backups, and monitoring and logging capabilities.[19]

OIT Support

PostgreSQL is not currently listed as an approved database technology by the FDOT Office of Information Technology.

Raster Data Support

PostGIS provides support for raster data via the *postgis_raster* extension. Direct support for reading and writing raster files in PostGIS is handled through the utilization of raster drivers from the Geographic Data Abstraction Library (GDAL).[20] GDAL is a separate open-source GIS software with a wide array of command-line utilities for working with geospatial data that is packaged with each PostGIS installation. Many open-source raster drivers are built into GDAL by default, and additional propriety drivers can be incorporated into GDAL by compiling the GDAL source code into a custom binary.

When combined in practice, the *postgis_raster* extension and the built-in capabilities of GDAL provide a large amount of functional and analytic power to work with raster data. A query against PostGIS v3.3.2.4975da8, which uses GDAL 3.0.4 as an integrated back-end component, returned 146 raster file formats and 288 raster functions supported by the default PostGIS 3.2.2 installation.

The *postgisaddons* repository, written by Pierre Racine, one of the original developers of the *postgis_raster* extension, provides additional raster support for performing common analytic operations against rasters.[21] These functions include generating area weighted summary statistics and performing either unions or buffered unions against multiple rasters. Tutorials and examples for using the *postgisaddons* functions can be found in a separate GitHub repository.[22]

---

[17] "Integrations," 2023
[18] "PostgreSQL/PostGIS – GDAL Documentation," 2023
[19] "Azure Database for PostgreSQL," 2023
[20] Rouault et al. "GDAL," 2023
[21] Racine "PostGIS Add-ons", 2023
[22] Ibid.

Rasters in PostGIS can be stored in two different formats:

- Native PostgreSQL/PostGIS rasters within the database
- Original, native format and registered "outdb" rasters with PostGIS in a file folder

Within-database rasters are stored within the PostgreSQL/PostGIS database as actual database tables. While there are performance gains to storing rasters this way, converting and loading large amounts of raster data into a PostgreSQL database requires a significant amount of time.

OutDB rasters, conversely, are stored on a local file system. The rasters are then registered in the PostgreSQL/PostGIS database server, which compiles metadata about the raster file for each within the PostGIS database. This metadata includes information related to the file path, coordinate system, geographic footprint, and more.

Once a raster is registered as an outdb raster in PostGIS, it can be used as if it were a raster stored natively within a database table. This has several advantages, the most immediate of which is that storing substantial amounts of raster data in its original file format within a file directory avoids the need to convert and load large amounts of data into the PostgreSQL database server. Outdb rasters impose a performance penalty when accessing and querying raster data in PostGIS, as it takes the database server additional time to find and read the original raster file into memory.

Lidar Point Cloud Support

PostgreSQL supports the storage of point cloud data in the form of binary large objects, or blobs. This allows users to store and retrieve point cloud data within a PostgreSQL database. In addition to storage, PostgreSQL also provides some basic functionality for working with point cloud data. For example, PostgreSQL includes the ST_Point cloud function, and this can be used to create a point cloud from a set of x, y, and z coordinates.

PostgreSQL also supports extensions such as pgPointCloud, providing users with additional point cloud functionality that involves point cloud compression, indexing, and querying. The pgPointCloud extension also has functions available to aid users in the importing and exporting process as well as geometric and spatial querying.[23]

Resource Description Framework Triples and Directed-Graph Data

PostgreSQL supports directed-graph data models and RDF triples via the Apache AGE extension.[24]

[23] "PostGIS – pgpointcloud," 2022
[24] "Apache AGE," 2023

3.2 Oracle Spatial

Oracle Database, a commercial RDBMS developed by the Oracle Corporation, provides a comprehensive database solution for managing and storing both spatial and non-spatial data. Today, Oracle Database is one of the most widely used commercial RDBMS in the world, and it has a reputation for being an exceedingly reliable and stable enterprise database platform. Oracle also provides commercial tools for managing database scalability, performance, and security in complex deployment environments.

FIGURE 5: ORACLE SPATIAL LOGO

The first version of Oracle Database (Oracle 2) was released in 1979 and at that time, Oracle was one of the first commercial RDBMS available on the market. The early versions of Oracle Database saw high levels of growth in large enterprise users in both the government and commercial markets with major revisions and upgrades for enterprise deployments released with Oracle 3 (1983) and Oracle 4 (1984). Today, Oracle is widely used in both the government and private industries, and the development is still active and ongoing, with Oracle Database 19c released in January of 2019.[25]

Oracle Database can be deployed on-premises, in the cloud, or in hybrid environments, giving organizations the flexibility to choose the deployment option that best suits their needs. It also offers seamless integration with Oracle Cloud Infrastructure (OCI), providing a range of cloud-based services for managing and scaling databases. Oracle also offers a variety of support and services to its customers, including 24/7 enterprise support, technical support, training, and consulting services.[26]

Oracle offers a variety of complex Enterprise Licensing Agreements (ELA) for licensing its software. One notable aspect to the Oracle ELA is that it contains provisions that allow the company to conduct license audits of a customer's IT infrastructure. The Oracle ELA also provides contingencies that allow Oracle to assess financial penalties if an audit reveals that a customer is utilizing Oracle's software to a greater extent than was originally licensed in the ELA. One side effect of Oracle's proactive stance towards customer audits has been the creation of a specialized market sector devoted to assisting Oracle customers in managing their license agreements and potential financial penalties associated with licensing violations.[27,28]

Oracle's spatial extension, Oracle Spatial and Graph, includes support for common geospatial data types as well as advanced features for spatial data analysis. It also allows for physical, logical, network, social, and semantic graph applications. The spatial features provide the schema and functionality needed to facilitate the storage, retrieval, update, and query of spatial feature collections in an Oracle database. The enterprise edition of Oracle Database is required for Spatial and Graph.

[25] "Oracle: A Company," 2002

[26] "Oracle Database," 2023

[27] "Oracle License Audit – Expert Advice," 2022

[28] "Oracle Software Audit Attorneys"

General Information	Oracle Spatial
License	Commercial
Cost	Commercial via FDOT ELA
Release Year	1979
Current Release	21c (January 2021)
Paid Enterprise Support	✓
OIT Supporte	½
RDF/Graph Data Model	Oracle Graph
Geospatial Support	
Vector Data	✓
Geometry	✓
Geometry Functions	
Geography	✓
Geography Functions	
Vector 3D & 4D	✓
3DM Support	✓
3DZ Support	✓
4DZM Support	✓
Topology & Network Data	✓
Coordinate Systems	✓
Custom Coordinate Systems	✓
Projections/Transformations	✓
Custom	✓
Raster Data	✓
Raster Functions	✓
Supported Raster Formats	✓
Lidar Point Cloud	✓
Extract-Transform-Load	
Native Spatial ETL Tools	✓
GDAL Compatible	✓
FME Compatible	✓

TABLE 2: ORACLE MATRIX

Geometry Support

Oracle's Spatial extension supports the 14 standard OGC SFA simple features, five extended SQL/MM Curve types, and two non-standard types that are specific to Oracle.[29]

Oracle supports several non-standard vector data types, in particular the solids and multi-solid collection types. Oracle refers to a solid as a geometry with multiple surfaces that is within a 3D environment.

Oracle Spatial and Graph has a large number of functions for working with vector data. It also includes various sets of specialized functions for tasks that are commonly performed by FDOT, such as referencing features to an LRS.

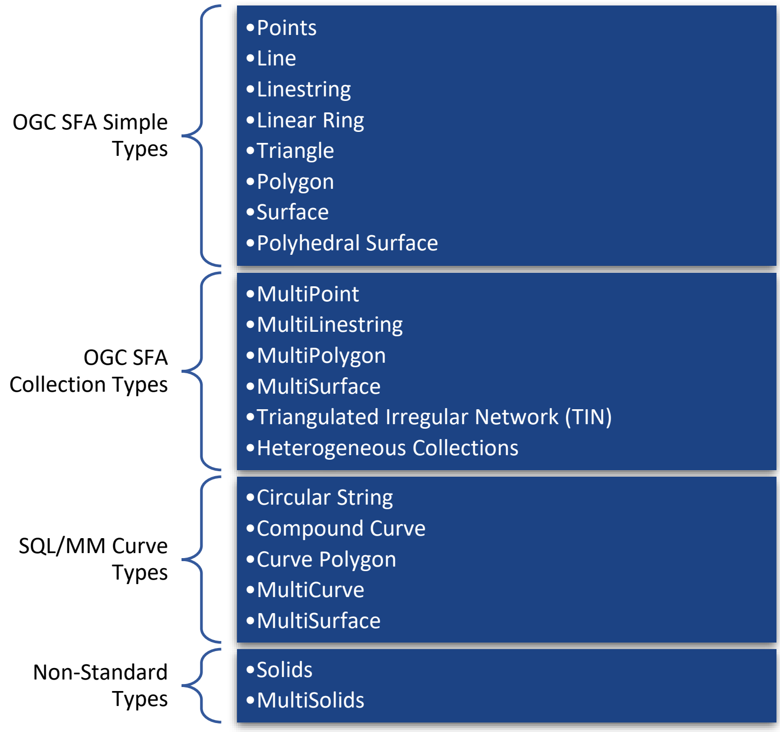

FIGURE 6: ORACLE VECTOR DATA TYPES

Geography Support

Oracle Spatial & Graph does not technically differentiate between a geography and geometry data type. The documentation for the current 19c release notes that with Oracle 9i, Spatial and Graph allows for work with geodetic coordinates. This implies that the spatial functions within Spatial and Graph automatically perform geodetic calculations when they encounter vector objects with geodetic coordinate systems, while cartesian calculations are made when they encounter vector objects with projected coordinates.[30]

The Spatial and Graph documentation also notes that some spatial functions utilize approximations when used with geodetic coordinate systems. Specifically, some functions will internally compute calculations in a local-tangent-plane Cartesian coordinate system before changing the results to the geodetic coordinate system. This type of approximation method could, in some cases, become a hidden source of real-world location error that would be extremely difficult to locate, as it is performed automatically and is not visible to the user.

3D & 4D Support

As of Oracle Spatial and Graph 19c, 3DM, 3DZ, and 4DZM objects are supported for all vector feature types.[31]

[29] "Spatial and Graph Developer's Guide," 2017
[30] Murray et al., "Coordinate Systems," 2023
[31] "Spatial and Graph Developer's Guide," 2017

Spatial and Graph supports referencing Z-elevation values to either an ellipsoid or a geoid.[32] Notably, the documentation notes that many spatial functions in Spatial and Graph do not support 3D features referenced against a geoid, and this is due to the additional computational complexity of calculating against an irregular surface. As with Oracle's behavior when working with the geography vector type, some of the functions use approximations methods for calculating results with 3D vector objects stored in geodetic coordinate systems.

Topology Support

Oracle Spatial and Graph includes the TOPOL module for creating, importing, storing, managing, and analyzing topology and network data. TOPOL also allows for the processing and analysis of data using Oracle Spatial Analysis functions. Oracle Spatial and Graph supports network analysis, such as shortest path and network routing analysis, through the use of network data.[33]

Support for Coordinate Systems and Transformations

Oracle Spatial and Graph provides support for custom coordinate systems, projections, and transformations through its Spatial Reference System (SRS) feature. Oracle Spatial and Graph includes a catalog of predefined coordinate systems, including well-known coordinate systems like WGS84 and UTM. Also, the Coordinate Transformation and Load (CTL) tool is available for use, enabling users to define custom projections and transformations for use in their spatial data. The CTL tool allows users to specify the source and target SRS as well as the parameters of the projection or transformation before saving it as a custom transformation in the database. Once defined, custom transformations can be used to convert spatial data between different coordinate systems or projections.

Oracle Spatial and Graph also includes a set of built-in transformation functions that support a wide range of standard coordinate systems and projections. These functions can be used for common spatial analysis tasks, such as converting between different projections, reprojecting spatial data, or calculating distances and areas.[34]

Extract Transform Load Support

Oracle Database provides several ETL tools that can be used to integrate and manage data within the database, and these involve the following:

- Oracle Warehouse Builder (OWB): OWB is a comprehensive ETL tool that enables users to extract, transform, and load data into a data warehouse. It provides a graphical interface for designing data integration workflows and supports integration with various data sources, including Oracle Database.[35]
- Oracle Data Integrator (ODI): ODI is a comprehensive data integration tool that supports ETL and ELT (Extract, Load, Transform) data integration scenarios. Similarly, ODI also has a graphical interface for the design of data integration workflows, allowing for integration with a series of different data sources, including Oracle Database.[36]

[32] Murray et al., "Spatial Concepts," 2023
[33] Murray et al., "Topology Data Model Overview," 2023
[34] Murray et al., "Coordinate Systems (Spatial Reference Systems)," 2023
[35] "Introduction to Oracle Warehouse Builder," 2010
[36] "Oracle Data Integrator," 2023

- Oracle SQL Developer: Oracle SQL Developer is a free, integrated development environment that includes ETL functionality. It supports the importing, exporting, profiling, and transformation of data using SQL or PL/SQL.[37]

FME Support

Oracle Spatial and Graph is compatible with FME, and it can be used as a data source or destination for FME workflows, allowing users to import or export geospatial data to and from an Oracle database. FME supports a wide range of geospatial data models, including Geographic Markup Language (GML), Geographic Javascript Object Notation (GeoJSON), Keyhole Markup Language (KML), allowing for data to be transferred to and from Oracle Spatial and Graph. FME provides a set of tools for manipulating and transforming geospatial data, such as reprojection, clipping, and attribute mapping. These tools can be used in conjunction with Oracle Spatial and Graph to perform complex data integration and transformation tasks.

By using FME with Oracle Spatial and Graph, users can easily integrate geospatial data from various sources and systems before transforming it into the desired format for analysis and visualization. This enhances the interoperability and flexibility of Oracle Spatial and Graph, making it a powerful tool for managing and analyzing geospatial data in a variety of contexts.[38]

GDAL Support

Oracle Spatial and Graph is compatible with GDAL, as it provides a GDAL driver that enables users to access and manipulate geospatial data stored in an Oracle database using GDAL tools and APIs. This driver supports both raster and vector data types, allowing users to perform a wide range of geospatial data processing tasks. By using the GDAL driver, users can easily integrate Oracle Spatial and Graph with other geospatial software tools that support GDAL, including QGIS, ArcGIS, and GRASS GIS. This makes it easier to work with geospatial data across different software platforms and tools.[39]

Overall, the compatibility with GDAL enhances the interoperability and flexibility of Oracle Spatial and Graph, positioning itself as a powerful tool for the management and analysis of geospatial data in a variety of contexts.

Azure RDS Support

To use Oracle database with Azure RDS, you can create a new instance of Azure RDS with Oracle Database as the database engine. Azure RDS supports several versions of Oracle Database, including Oracle Database 19c, 18c, and 12c.

OIT Support

Oracle is not currently listed as an approved database platform technology by the FDOT Office of Information Technology.

Raster Data Support

In Oracle Spatial and Graph, raster data is stored in a specialized format called GeoRaster. GeoRaster data is stored in a table that has a column with the SDO_GEORASTER data type. This column contains a binary

[37] "SQL Developer," 2023
[38] "Integrations," 2023
[39] "Oracle Spatial – GDAL Documentation," 2023

representation of the raster data and metadata, including the size, resolution, and CRS of the raster. The metadata is stored as an XML file, allowing for it to be easily parsed and queried.

GeoRaster supports various types of raster data, including single-band images, multi-band images, and raster datasets with varying pixel sizes and resolutions. It also provides support for compression and tiling, which can help to reduce storage requirements and improve query performance. Oracle Spatial and Graph provides a number of tools for working with GeoRaster data, including APIs for loading and querying raster data and other tools for visualizing and analyzing raster data. Additionally, Oracle Spatial and Graph supports a variety of industry-standard data formats for importing and exporting raster data, including GeoTIFF, JPEG2000, and MrSID.[40]

Some examples of the raster functions available in Oracle Spatial and Graph include:

- Georeference: Used to define the spatial reference system for a raster image.
- Reproject: Allows for reprojection of a raster image from one coordinate system to another.
- Crop: Helps user to crop a raster image to a specified extent.
- Raster Mosaic: Combines multiple raster datasets into a single dataset.
- Raster Re-Sample: Changes the resolution of a raster dataset.
- Raster Clip: Extracts a subset of a raster dataset.
- Raster Slope and Aspect: Calculates the slope and aspect of a raster dataset.
- Raster Distance and Proximity: Calculates the distance and proximity of raster cells to a specified location.
- Raster Zonal Statistics: Calculates statistical information for each zone of a raster dataset.
- Raster Terrain Analysis: Performs various terrain analysis tasks on a raster dataset, such as hill-shading and viewshed analysis.[41]

Lidar Point Cloud Support

Lidar data can be stored in Oracle Spatial and Graph using the GeoRaster data type. GeoRaster supports the storage of lidar data as raster data, as each point of the cloud is stored as an individual pixel within the raster dataset. This approach allows for the efficient storage and querying of lidar data while also providing the ability to perform raster-based analysis operations.

To store lidar data in Oracle, the lidar point cloud data can be converted to raster format and then stored in a GeoRaster column within an Oracle table. The conversion process can be done using various software tools, including Oracle's own 3D lidar processing tools or third-party software like FME or ArcGIS. Once the lidar data is stored as a GeoRaster, it can be queried and analyzed using Oracle Spatial and Graph's built-in functions and tools. For example, users can perform elevation-based analysis with the support of built-in raster functions, such as slope and aspect calculations, or point-in-polygon analysis by overlaying the raster data with polygonal boundaries.

In addition to storing lidar data as raster data, Oracle Spatial and Graph also provides support for the storage and management of lidar data as point cloud data through the use of the SDO_PC data type. This

---

[40] "GeoRaster in Oracle Database," 2021

[41] Murray et al., "Preface," 2023

approach allows users to store and manage the raw lidar data as point cloud data, enabling enhanced analysis and visualization of the lidar data.

Oracle Spatial and Graph provides a variety of lidar functions that can be used for analyzing, processing, and managing lidar data. Some of the key lidar functions that are supported in Oracle Spatial and Graph include the following:

- Point Cloud Loading and Indexing: Oracle Spatial and Graph supports loading lidar point cloud data using the open LAS interchange format and allows for the indexing of data for efficient querying and analysis.
- Point Cloud Processing: Oracle Spatial and Graph provides a range of point cloud processing functions, including filtering, interpolation, and classification of lidar point clouds.
- Point Cloud Visualization: Oracle Spatial and Graph provides several tools for visualizing lidar point clouds, including 2D and 3D viewers that allow users to view, rotate, and zoom in on lidar data.
- Point Cloud Analytics: Oracle Spatial and Graph provides a variety of lidar analytics functions, including point-in-polygon analysis, height analysis, and change detection analysis.
- Point Cloud Export: Oracle Spatial and Graph supports exporting lidar point cloud data to various formats, including LAS and ASCII formats, giving users the ability to easily share and exchange lidar data with other applications and groups.[42]

Overall, these lidar functions enable users to perform a wide range of analysis and processing tasks on lidar data stored in Oracle databases, resulting in a powerful platform for managing and analyzing lidar data.

Resource Description Framework Triples and Directed-Graph Data

Oracle supports directed-graph data models and RDF triples via the Oracle Graph extension.[43]

[42] Murray et al., "SDO_PC_PKG Package," 2023
[43] "Oracle Graph," 2023

3.3 Microsoft SQL Server

Microsoft SQL Server is a RBDMS that was first developed as a 16-bit relational database for the OS/2 operating system, and it was originally released as Sybase SQL Server in 1989. In 1992, Microsoft released its own version of SQL Server for Windows NT, marking their first entry into the enterprise database market in competition against the likes of Oracle, IBM, and others major enterprise software vendors. Continued investment from Microsoft and a focus on developing a scalable and cost-effective platform for enterprise applications helped SQL server grow in popularity with businesses and organizations of all sizes in the following years. Microsoft SQL server is still in active development today, with the release of SQL Server 2022 in November of 2022. Currently, SQL Server holds over 31% of the enterprise database market share[44], and it is utilized by more than 95% of Fortune 500 companies.[45]

FIGURE 7: MICROSOFT SQL SERVER LOGO

Microsoft provides a variety of different commercial licenses for SQL Server, ranging from single-computer installations to massive public-facing applications with many concurrent users. These licenses, called editions in Microsoft's documentation, include Enterprise, Standard, Web, Developer, and Express. Each edition of SQL Server differs in scope and pricing, with features varying extensively between the free, entry-level Express edition and the premium Enterprise deployment with unlimited virtualization capability.[46] Microsoft also offers the ability deploy SQL Server within the Azure cloud platform. This can be done as a managed platform as a service (PaaS) with Azure SQL Database or with SQL Server on Azure Virtual Machines as an infrastructure as a service (IaaS).[47]

[44] "Microsoft SQL Server - Market Share, Competitor Insights in Database," 2023
[45] "Microsoft Inspire," 2022
[46] rwestMSFT. "Editions and Supported Features of SQL Server 2022 - SQL Server," 2023
[47] MashaMSFT. "What is Azure SQL?," 2023

General Information	SQL Server
License	Commercial
Cost	Commercial via FDOT ELA
Release Year	April 24, 1989
Current Release	SQL Server 2022 (November 16,2022)
Paid Enterprise Support	✓
OIT Supported	✓
RDF/Graph Data Model Support	SQL Graph
Geospatial Support	
Vector Data	✓
Geometry	✓
Geometry Functions	
Geography	✓
Geography Functions	
Vector 3D & 4D	✓
3DM Support	Read/Write Only
3DZ Support	Read/Write Only
4DZM Support	Read/Write Only
Topology & Network Data	✗
Coordinate Systems	✗
Custom Coordinate Systems	✗
Projections/Transformations	✗
Custom	✗
Raster Data	✗
Raster Functions	✗
Supported Raster Formats	BLOB Only
Lidar Point Cloud	✗ BLOB Only
Extract-Transform-Load	
Native Spatial ETL Tools	✓
GDAL Compatible	✓
FME Compatible	✓

TABLE 3: SQL SERVER MATRIX

Geometry Support

Microsoft SQL Server supports seven OGC SFA vector data types and three SQL/MM extended geometry types. SQL Server also allows for five additional standard geometry types that are "not instantiable," meaning they are used behind the scenes but cannot be directly created or manipulated within a SQL Server database.[48]

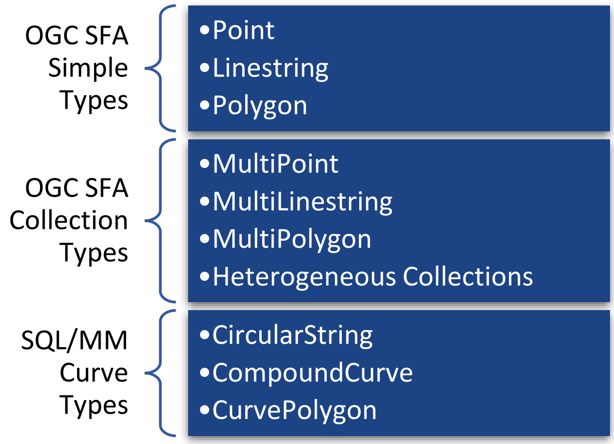

FIGURE 8: SQL SERVER VECTOR DATA TYPES

SQL Server has over 70 different spatial functions that support using geometry data types as an input parameter. These functions range from basic geometric operations, such as calculating distance and area, to more advanced operations like buffer analysis, spatial intersection, and spatial aggregation. Some spatial functions and methods in SQL Server only work with simple geometry types and cannot be used with collections.[49] It is worth noting that SQL Server's library of functions for working with geometry data is significantly smaller and less capable than that of PostgreSQL/PostGIS or Oracle Spatial & Graph.

Microsoft has released *SQL Server Spatial Tools* as a separate and optional add-on to the standard spatial functions that are available in the default SQL Server installation.[50] This optional extension for SQL Server contains additional spatial functions that can be used for specialized tasks, including linear referencing.

Geography Support

The Microsoft SQL Server geography data type supports the same ten OGC geometry data types as the geometry data type with the addition of the *FullGlobe* data type. This additional option is a distinct type of polygon that covers the entire globe with an area but no borders or vertices.[51]

3D & 4D Support

Microsoft SQL Server has the ability to define and access M and Z values for both geographies and geometries, but it does not share the same capabilities as PostgreSQL/PostGIS and Oracle for working with 3DM, 3DZ or 4DZM vector objects. The SQL Server documentation states that Z and M values are read only and are not used in any spatial functions.[52]

Topology Support

Microsoft SQL Server does not support topology or topology-aware vector features.

---

[48] MladjoA. "Spatial Data Types Overview - SQL Server," 2023
[49] Ibid.
[50] "Microsoft SQL Server Spatial Tools," 2019
[51] MladjoA. "Spatial Data Types Overview - SQL Server," 2023
[52] MladjoA. "M (Geometry Data Type) - SQL Server," 2023

Support for Coordinate Systems and Transformations

Microsoft SQL Server supports spatial reference identifiers (SRID) for both geometry and geography spatial instances. The SRID corresponds to a Spatial Reference System (SRS) based on the specific ellipsoid used for either flat-earth or round-earth mapping.[53] Spatial columns can have objects with different SRIDs, but the spatial instance being compared must have the same SRID for the spatial methods of SQL Server to function properly. SQL Server does not natively offer the ability to transform data from one spatial reference to another, although there are third party solutions that offer aspects of this functionality as add-ons.

Extract Transform Load Support

There are variety of processes and tools used for ETL steps that are available with Microsoft SQL Server. When setting up SQL Server, end-users can install SQL Server Integration Services (SSIS), a platform for building enterprise-level ETL processes for data integration and transformation. Integration services can extract and transform data from a variety of sources, including XML files, flat files, and relational data sources. Then, the data can be loaded into one or multiple destinations.[54] The Integration Services platform includes both programmatic and graphical tools for building ETL packages as well as the ability to store, run, and manage these packages.

SQL Server Management Studio (SSMS) provides the native Import and Export Data Wizard, which uses SSIS in the background. This provides users with a straightforward way to ingest data into their SQL Server instance. The Import and Export Data Wizard can copy files to and from Enterprise databases, including other SQL Server instances, Oracle, DB2, and more. Text files (flat files), Microsoft Excel and Microsoft Access files, Azure Blob Storage, open-source databases (PostgreSQL, MySQL, etc.), and other data sources have drivers available for connecting to SQL Server. These sources require an available ODBC driver, .Net framework provider, or OLE DB provider.[55]

FME Support

FME offers native support for SQL Server, allowing users to easily connect to SQL Server databases and work with data stored within them. FME can read and write data to SQL Server, and it can support SQL Server Spatial data types, making it an excellent choice for spatial data integration projects. Additionally, FME's support for SQL Server allows users to leverage its advanced data processing capabilities, involving stored procedures, triggers, and views, as part of their data integration workflows. [56]

GDAL Support

Microsoft SQL Server is GDAL-compatible with the 'MSSQLSpatial' driver available which provides access to spatial tables with geometry columns.[57]

Azure RDS Support

Microsoft provides Azure SQL Database as a fully managed PaaS solution. Most of the behind-the-scenes technical management functionality like upgrading, patching, and monitoring is handled without end-user

---

[53] MladjoA. "Spatial Reference Identifiers (SRIDs) - SQL Server," 2023

[54] "SQL Server Integration Services - SQL Server Integration Services (SSIS)," 2023

[55] "Import and Export Data with a SQL Server Import and Export Wizard," 2023

[56] "Integrations," 2023

[57] "MSSQLSpatial," 2023

development and the database is kept current on the latest stable version of SQL Server with extremely high availability with Microsoft handling the underlying infrastructure.[58]

OIT Support

FDOT currently utilizes SQL Server, and the software is supported and approved by the Office of Information Technology (OIT). SQL Server 2016, SQL Server 2017, SQL Server 2019, Azure SQL, and Azure SQL Managed instance are all listed as supported database platforms.

Raster Data Support

Microsoft SQL server does not natively support raster data types, but users can store images as binary large object (BLOB) data. Additionally, there are no functions in Microsoft SQL Server to work with raster data types. Users can, however, leverage third party software, such as GDAL (Geospatial Data Abstraction Library), to help process raster data.

Lidar Point Cloud Support

Microsoft SQL Server does not natively support point cloud data types or provide specific functions for working with point clouds. However, it is possible to work with point cloud data in SQL Server by storing the point cloud data as binary data.

To work with point cloud data in SQL Server, users can leverage third-party tools or libraries that support point cloud data. These include options like the Point Data Abstraction Library (PDAL), Potree, or Entwine. These tools can convert point cloud data into other formats, such as CSV or JSON, which can be loaded into SQL Server tables.

Resource Description Framework Triples and Directed-Graph Data

SQL Server supports directed-graph data models and RDF triples via the SQL Graph extension.[59]

---

[58] WilliamDAssafMSFT. "What is Azure SQL Database Service?," 2023
[59] "SQL Graph," 2023

3.4 Snowflake

Snowflake is a cloud-based data warehousing company that provides a fully managed, scalable, and secure platform for storing, processing, and analyzing large volumes of structured and semi-structured data. Snowflake was originally founded in 2012 by data warehousing experts. The founders of Snowflake saw an opportunity to disrupt the data warehousing market by leveraging the cloud to provide a more scalable and cost-effective solution for the storage and analysis of large volumes of data. Snowflake has developed a new architecture called the "Snowflake Elastic Data Warehouse," and it breaks data computation and data storage into separate, discretely managed elements. Today, Snowflake is one of the largest and most mature companies with offerings in the "data-as-a-service" market.[60]

FIGURE 9: SNOWFLAKE LOGO

Snowflake launched its cloud-based data warehouse service in 2015, and it quickly gained traction among enterprise customers. In 2018, Snowflake raised $450 million in a funding round led by Sequoia Capital, becoming one of the most valuable privately held companies in the world. Snowflake went public on the New York Stock Exchange in September of 2020, and its initial public offering (IPO) was one of the largest software IPOs in history, raising $3.4 billion. Today, Snowflake is used by thousands of customers worldwide, including some of the world's largest and most well-known companies in industries such as healthcare, finance, retail, and technology.[61,62]

Snowflake's architecture allows for the separation of storage and computation which helps organizations to scale each independently and according to need. This capability lets organizations store and access substantial amounts of data without being concerned about the performance impact of processing large queries on the overall storage solution. Snowflake's cloud-based platform also allows users to scale their storage and computation resources up or down as needed, making it easy to allocate and deallocate cloud resources to match changing business needs. Snowflake also provides robust security features, including multi-factor authentication, data encryption, and access controls, and it is compliant with a wide range of regulatory requirements, including SOC 2, PCI DSS, and HIPAA.[63]

[60] Handy, A. "Snowflake offers cloud data warehouse as a service," 2013
[61] "Snowflake raises $450 million in another VC round," 2018
[62] Monica, P. R. L. "Snowflake shares more than double," 2020
[63] "Introduction to Snowflake," 2023

General Information	Snowflake
License	Commercial
Cost	Priced by Usage
Release Year	October, 2014
Current Release	N/A
Paid Enterprise Support	✓
OIT Support	½
RDF/Graph Data Model Support	✗
Geospatial Support	
Vector Data	✓
Geometry	✓
Geometry Functions	
Geography	✓
Geography Functions	
Vector 3D & 4D	✓
3DM Support	✓
3DZ Support	✗
4DZM Support	✗
Topology & Network Data	✗
Coordinate Systems	✗
Custom Coordinate Systems	✗
Projections/Transformations	✗
Custom	✗
Raster Data	✗
Raster Functions	✗
Supported Raster Formats	✗
Lidar Point Cloud	✗
Extract-Transform-Load	
Native Spatial ETL Tools	✓
GDAL Compatible	
FME Compatible	✓

TABLE 4: SNOWFLAKE MATRIX

Geometry Support

In Snowflake, geometries are an essential component of the geospatial data type, which allows users to store and analyze geographic data within the Snowflake data warehouse. With the growing importance of location-based analysis in today's data-driven world, understanding geometries and their use in Snowflake is crucial for anyone looking to derive insights from spatial data.

Snowflake supports a GeoJSON data types in the standard Well-Known Text (WKT), Well-Known Binary (WKB), Extended Well-Known Text (EWKT) and Extended Well-Known Binary (EWKB) formats, including[64]:

- WKT/WKB/EWKT/EWKB GeoJSON Geospatial Objects
  - Point
  - Multipoint
  - LineString
  - MultiLineString
  - Polygon
  - MultiPolygon
  - GeometryCollection

- GeoJSON-Specific Geospatial Objects
  - Feature
  - FeatureCollection

Snowflake documentation currently lists 48 spatial functions for use with geometries. Uses include conversion, parsing, transformation, and relation and measurement functions.[65]

Geography Support

Snowflake provides geospatial functions that enable users to perform location-based analyses and gain insights into their data. These functions allow for geospatial data to be stored, queried, and analyzed within the Snowflake data warehouse.

Snowflake supports more spatial functions that are available to be used with geographies. The ST_INTERSECTION and ST_UNION functions are not available to geometry data types within the Snowflake environment. A query of the number of spatial functions available to the geography data type reveals a total of 56.[66]

3D & 4D Support

Snowflake does not support 3D or 4D data types for either geometry vector objects or geography vector objects.[67]

Topology Support

Snowflake does not support topology data types.

---

64 "Geospatial Data Types | Snowflake Documentation," 2023

65 "Geospatial Functions | Snowflake Documentation," 2023

66 Ibid.

67 "Geospatial Data Types| Snowflake Documentation," 2023

Support for Coordinate Systems and Transformations

Snowflake has limited support for coordinate systems and does not support transformations.[68]

Extract Transform Load Support

Snowflake offers several ETL tools and services that allow users to integrate and transform data from various sources to fit their analytical needs. These tools include data ingestion services, data transformation services, and data loading services. Some of the ETL tools and services offered by Snowflake include the following:

- Snowflake Data Pipelines: Snowflake Data Pipelines is a fully managed service that allows users to easily build, run, and manage data pipelines that integrate data from various sources, including databases, files, and cloud services, and load the data into Snowflake. The service includes pre-built connectors for popular data sources, such as Amazon S3, Azure Blob Storage, and Google Cloud Storage, and it allows users to transform the data using SQL or Python.

- Partner ETL integrations: Snowflake has partnerships with several ETL vendors that offer pre-built connectors and options for integration. These ETL tools allow users to integrate data from multiple sources, transform the data using visual or code-based interfaces, and load the data into Snowflake.

- Snowflake Connector for Spark: The Snowflake Connector for Spark allows users to integrate data from Spark-based data processing platforms, such as Databricks and Apache Spark, and load the data into Snowflake. The connector provides a high-performance, parallelized interface for loading large amounts of data into Snowflake.[69]

FME Support

As of FME Desktop version 2020.0 or above, FME can interact with Snowflake databases using the Snowflake Reader and Writer FME Tools to read and write data into any other FME supported format.[70] Additionally, FME versions 2020.1 and above can read and write geospatial data to and from Snowflake with the use of the Snowflake Spatial Reader and Writer FME tools. This data can be stored as WKB, WKT, and GeoJSON, and the geography data type supports points, lines, and polygons.[71] FME also allows a user to access the native Snowflake geospatial functions with either the SQLCreator or SQLExecutor on the Reader "where clause" or through the FeatureReader "where clause" itself.

GDAL Support

GDAL does not currently list supported vector or raster drivers for Snowflake. Snowflake does support connecting using 3rd party client services and applications that support ODBC (Open Database Connectivity) and developing applications through snowflake connectors/drivers for Python, Node.js, Spark, etc. However, Connecting using these methods does require additional installation, configuration, and development.[72] GDAL does support ODBC connections for vector-based data and offers a Python API.

---

[68] "Geospatial Data Types | Snowflake Documentation," 2023
[69] "Overview of the Spark Connector," 2023
[70] "Using FME and the Snowflake Database Format," 2022
[71] "Using Snowflake Spatial in FME for Geospatial Data," 2022
[72] "Logging into Snowflake," 2023

Azure RDS Support

Snowflake can be used with Azure RDS by leveraging the Snowflake Connector for Python, which allows users to connect to Snowflake using Python-based applications, including those running on Azure RDS. The Snowflake Connector for Python can be installed on an Azure RDS instance by following the installation instructions provided by Snowflake. Once the connector is installed, users can use Python-based applications to read and write data to Snowflake. This can be useful for tasks such as extracting data from Snowflake and loading it into an Azure RDS database, or for analyzing data stored in Snowflake using Python-based tools and libraries.[73]

OIT Support

Snowflake is not currently listed as a supported database platform technology by the FDOT Office of Information Technology.

Raster Data Support

Snowflake does not offer native support for raster data types. While Snowflake does provide robust support for vector data types, users looking to store and analyze raster data will need to employ alternative solutions or workarounds, such as converting raster data to vector format or using third-party tools for raster processing.[74]

Lidar Point Cloud Support

Snowflake does not offer native support for point cloud data types. However, users can work with third-party tools or libraries that support point cloud data, such as PDAL to convert point cloud data into formats that can be loaded into Snowflake tables. Once loaded, users can leverage Snowflake's powerful SQL querying capabilities and spatial functions to analyze and visualize the data.

Resource Description Framework Triples and Directed-Graph Data

Snowflake does not support directed-graph data models or RDF triples.

[73] "Snowflake on Microsoft Azure," 2023

[74] "Unsupported Data Types | Snowflake Documentation," 2023

3.5 Neo4j

Neo4j is a graph database management system that, unlike traditional relational databases, stores data in a graph format rather than as a set of tables related to each other through foreign keys. Graph databases store data as nodes, which represent entities and edges. Edges refer to the relationships and connections between and among entities, and these entities allow for more efficient querying and analysis of complex data structures that might not be amenable, compatible, or representable within the framework of more traditional RDMS. Neo4j is capable of handling both structured and unstructured data, allowing for a diverse set of data types to be stored and queried within it.

FIGURE 10: NEO4J LOGO

Graph databases are a relatively recent technology, and while many of the companies and offerings on the market are new and immature, neo4j can be considered one of the originals, as it is the most well-established company with the most mature database product in the market.

Initially founded in 2007, neo4j raised $2.5m in 2009 and made its first product offering publicly available in 2010, with the release of neo4j v1.0. Soon after the release of its v1.0 product, the company underwent a series of large, successful rounds of venture capital investment, most recently with a successful $325m Series F round in late 2021.[75,76] As both a company and database technology, neo4j is considered to be a market leader within the rapidly growing graph-database technology space. Neo4j is in active development, as neo4j v5.6 was recently released on March 24, 2023.

Neo4j uses the query language Cypher, a proprietary language inspired by SQL. Neo4j markets Cypher as a query language that allows users to write complex, highly optimized queries for graph-based data quickly and easily, translating common “node-edge” search patterns into a human-readable format. Cypher is exclusive to the neo4j databases, and the query language is not cross-compatible with the widely used SQL standard.

[75] “Neo4j -Funding, Financials, Valuation & Investors,” 2023
[76] “Neo4j Announces $325 Million Series F Investment,” 2021

General Information	neo4j
License	Open-Source Community Edition Commercial
Cost	Free (Community Edition) Commercial Edition
Release Year	February 2010 (v1.0)
Current Release	v5.6 (March 2023)
Paid Enterprise Support	✓
OIT Support	✗
RDF/Graph Data Support	✓
Geospatial Support	
Vector Data	✗
Geometry	Point Geometries Only
Geometry Functions	3
Geography	Point Geographies Only
Geography Functions	3
Vector 3D & 4D	
3DM Support	✗
3DZ Support	Point Geographies/Geometries Only
4DZM Support	✗
Topology & Network Data	✓
Coordinate Systems	✗
Custom Coordinate Systems	✗
Projections/Transformations	✗
Custom	✗
Raster Data	✗
Raster Functions	✗
Supported Raster Formats	✗
Lidar Point Cloud	✗
Extract-Transform-Load	
Native Spatial ETL Tools	✗
GDAL Compatible	✗
FME Compatible	JDBC Only

TABLE 5: NEO4J MATRIX

Geometry Support

neo4j only supports the point vector type, which can have either 2D or 3D coordinates.[77] Instances of *point* and *list of point* can be assigned to node and relationship properties, and nodes with these properties can be indexed using a point index. The distance function can be applied using any 2D or 3D CRS if the points being referenced utilize the same CRS.[78]

FIGURE 11: NEO4J SUPPORTED VECTOR DATA TYPES

Geography Support

Neo4j supports four Coordinate Reference Systems (CRS). Of these four, two use geographic coordinates to model points on earth, while the other two rely on cartesian coordinates to model points in space.[79]

- Geographic Coordinate Systems
  - WGS-84: longitude, latitude (x,y)
  - WGS-84-3D: longitude, latitude, height (x,y,z)
- Cartesian Coordinate Systems
  - Cartesian: (x, y)
  - Cartesian 3D: (x,y,z)

3D & 4D Support

Neo4j supports the assignment of Z values for point geometries with the *point()* function, but it only offers basic distance and bounding box calculations for working with 3DZ point data.[80] Instances of *point* and *list of point* can be assigned to node and relationship properties, and nodes with these properties can be indexed using a point index. The distance function can be applied using any 2D or 3D CRS provided that the point uses the same CRS.[81]

Topology Support

Neo4j supports the topological sorting of nodes in the graph where every node appears only after all the nodes pointing to it have appeared. The topological order of nodes is only defined for directed acyclic graphs (DAGs).[82] The topological sort is also listed as "alpha tier," indicating it is an experimental feature that may be changed or removed at any time.[83]

---

[77] "Spatial Values," 2023
[78] Ibid.
[79] Ibid.
[80] "Spatial Functions," 2023
[81] "Spatial Values," 2023
[82] "Topological Sort," 2023
[83] "Introduction – Neo4j Graph Data Science," 2023

Support for Coordinate Systems and Transformations

Neo4j only offers four coordinate reference systems, with two geographic coordinate reference systems (WGS-84 and WGS-84-3D) and two Cartesian coordinate reference systems (Cartesian and Cartesian 3D). These CRSs cannot be compared or implicitly converted from one to the other.[84]

Extract Transform Load Support

The ETL process with neo4j is slightly different than with other RDBMS due to its inherent structural and schematic setup. Neo4j offers native ETL tooling to convert from relational database data into the neo4j graph database format. A simplified description of the process involves setting up a database connection to an existing relational database, selecting the Neo4j database import location, verifying the created schema mapping and graph data model, and running the ETL tool.[85]

The tooling provides the following JDBC drivers natively to connect to existing relational databases: PostgreSQL, MySQL, db2, and JDBC. When the relational data is translated into graph data, there are three rules that the tool utilizes to make the change, and these are outlined below[86]:

- A table with a foreign key is treated as a join and imported as a node with a relationship.
- A table with two foreign keys is treated as a join table and imported as a relationship.
- A table with greater than two foreign keys is treated as an intermediate node and imported as a node with multiple relationships.

Neo4j also offers integrations and support for Enterprise customers for Apache Spark, Kafka, and Business Intelligence. It also offers the ability to load flat csv files with the LOAD CSV Cypher command. CSV files are loaded with the rows as individual nodes, and the columns set as properties on the roads.

FME Support

Connecting to FME requires the use of a JDBC (Java Database Connectivity) API driver. After acquiring a JDBC driver and placing it in the appropriate folder, users can create a database connection within FME using the connection string provided by the JDBC driver. FME also supports the ability to add connection properties (such as 'flatten = -1' to ensure that all data is returned from variable-schema databases like neo4j). Users can also use FME to interact with the data in a neo4j database, but this is completed in a different manner than relational databases. Instead of using a reader or writer, end-users can utilize SQLCreator or SQLExecutor transformers to run native Cypher language queries and return data into the FME environment.[87]

GDAL Support

GDAL does not currently list supported vector or raster drivers for Neo4j.[88]

---

[84] "Spatial Values," 2023
[85] "How-To: Neo4j ETL Tool - Developer Guides," 2023
[86] Ibid.
[87] "JDBC: The Swiss Army Knife of Database Formats," 2022
[88] Rouault et al. "GDAL," 2023

Azure RDS Support

The enterprise edition of Neo4j is deployable on Azure virtual machines through Azure Marketplace either as a single instance or causal cluster.[89]

OIT Support

Neo4j is not currently listed as a supported database platform technology by the FDOT Office of Information Technology.

Raster Data Support

Storing binary raster data in neo4j is not particularly efficient, but different implementations have been developed where the raster data is stored on disk. The neo4j database contains metadata, including search trees (R-trees in neo4j nomenclature) to provide relatively quick access to raster datasets.[90]

Lidar Point Cloud Support

Neo4j does not currently support the point cloud data types or provide specific functions working with point clouds.

Resource Description Framework Triples and Directed-Graph Data

Neo4j is a graph database and supports directed-graph data models and RDF triples natively.

[89] "Hosting Neo4j in the Cloud," 2023

[90] Hein, N., & Blankenbach, J. "Evaluation of a NoSQL Database for Storing Big Geospatial Raster Data," 2021

[THIS PAGE INTENTIONALLY LEFT BLANK]

6. REFERENCES

A Brief History of PostgreSQL. (2023, February 9). PostgreSQL Documentation. https://www.postgresql.org/docs/15/history.html

Apache AGE. (2023, June 1). https://age.apache.org/

ArcGIS 11.1 and ArcGIS Pro 3.1 requirements for Microsoft SQL Server—ArcGIS Enterprise | Documentation for ArcGIS Enterprise. (2023, April 25). https://enterprise.arcgis.com/en/system-requirements/latest/windows/database-requirements-sqlserver.htm

Azure Database for PostgreSQL | Microsoft Azure. (2023, April 25). https://azure.microsoft.com/en-us/products/postgresql/

Azure SQL Database: Grow your potential in the cloud. (2023, April 25). https://www.stitchdata.com/resources/azure-sql-database/

Chapter 4. Data Management. (2023, April 25). https://postgis.net/docs/using_postgis_dbmanagement.html#OGC_Geometry

Chapter 8. PostGIS Reference. (2023, April 25). https://postgis.net/docs/manual-3.2/reference.html

Chapter 10. Topology. (2023, April 25). https://postgis.net/docs/Topology.html

Chapter 15. PostGIS Special Functions Index. (n.d.). Retrieved May 10, 2023, from https://postgis.net/docs/PostGIS_Special_Functions_Index.html#PostGIS_TypeFunctionMatrix

Companies That Use PostgreSQL - PostgreSQL Clients. (2023, April 25). *Thomson Data*. https://www.thomsondata.com/customer-base/companies-that-use-postgresql.php

Connect to Oracle from ArcGIS—ArcGIS Pro | Documentation. (2023, April 25). https://pro.arcgis.com/en/pro-app/latest/help/data/databases/connect-oracle.htm

Connect to Snowflake from ArcGIS—ArcGIS Pro | Documentation. (2023, April 25). https://pro.arcgis.com/en/pro-app/latest/help/data/databases/connect-snowflake.htm

Copy: PostgreSQL. (2023, February 9). PostgreSQL Documentation. https://www.postgresql.org/docs/15/sql-copy.html

Cost Estimating Guide for Road Construction. (n.d.).

Cost Per Mile Models Reports. (2023, April 25). FDOT. https://www.fdot.gov/programmanagement/estimates/documents/costpermilemodelsreports

Create a geodatabase in PostgreSQL on Windows—ArcGIS Pro | Documentation. (2023, April 25). https://pro.arcgis.com/en/pro-app/latest/help/data/geodatabases/manage-postgresql/setup-geodatabase-postgresql-windows.htm

Database connections in ArcGIS Pro—ArcGIS Pro | Documentation. (2023, April 25). https://pro.arcgis.com/en/pro-app/latest/help/data/databases/database-connections-in-arcgis-pro.htm

DB-Engines Ranking—Method. (2023, April 25). https://db-engines.com/en/ranking_definition

EDB Enterprise Plan. (2023, April 25). EDB. https://www.enterprisedb.com/products/edb-enterprise

Elephant versus Dolphin: Which is Faster? Which is Smarter? - Redfin Real Estate News. (2023, April 25). https://www.redfin.com/news/elephant_versus_dolphin_which_is_faster_which_is_smarter/

Evenden, Gerald I., Rouault, Even, Warmerdam, Frank, Evers, Kristian, Knudsen, Thomas, Butler, Howard, Taves, Mike W., Schwehr, Kurt, Sales de Andrade, Elliott, Karney, Charles, Couwenberg, Sebastiaan, Dawson, Nyall, Snow, Alan D., & Jimenez Shaw, Javier. (2023). *PROJ* (9.2.0). Zenodo. https://doi.org/10.5281/ZENODO.5884394

FHP: Surveyor Killed By I-95 Driver Who Tried Passing Slower Traffic. (2016, April 29). https://www.cbsnews.com/miami/news/fhp-surveyor-killed-by-i-95-driver-who-swerved-to-avoid-another-car/

GeoRaster in Oracle Database. (2021).

Geospatial Data Types | Snowflake Documentation. (2023, April 25). https://docs.snowflake.com/en/sql-reference/data-types-geospatial

Geospatial Functions | Snowflake Documentation. (2023, April 25). https://docs.snowflake.com/en/sql-reference/functions-geospatial

Handy, A. (2014, October 23). Snowflake offers cloud data warehouse as a service, cheaply. SD Times. https://sdtimes.com/amazon-web-services/snowflake-offers-cloud-data-warehouse-service-cheaply/

Hein, N., & Blankenbach, J. (2021). Evaluation of a NoSQL Database for Storing Big Geospatial Raster Data. *GI_Forum*, *1*, 76–84. https://doi.org/10.1553/giscience2021_01_s76

Hosting Neo4j in the Cloud—Developer Guides. (2023, April 25). Neo4j Graph Data Platform. https://neo4j.com/developer/guide-cloud-deployment/

How-To: Neo4j ETL Tool—Developer Guides. (n.d.). Retrieved May 10, 2023, from https://neo4j.com/developer/neo4j-etl/

IFC - Industry Foundation Classes/IFC classes—Wiki.OSArch. (2023, April 25). https://wiki.osarch.org/index.php?title=IFC_-_Industry_Foundation_Classes/IFC_classes

IFC Formats. (2023, April 25). BuildingSMART Technical. https://technical.buildingsmart.org/standards/ifc/ifc-formats/

Import and Export Data with the SQL Server Import and Export Wizard—SQL Server Integration Services (SSIS). (2023a, February 28). https://learn.microsoft.com/en-us/sql/integration-services/import-export-data/import-and-export-data-with-the-sql-server-import-and-export-wizard

Industry Foundation Classes (IFC). (2023, April 25). BuildingSMART Technical. https://technical.buildingsmart.org/standards/ifc/

Integrations. (2023, April 25). FME by Safe Software. https://fme.safe.com/solutions/integrations/

Introduction to Oracle Warehouse Builder. (2023, April 25). https://docs.oracle.com/cd/E18283_01/owb.112/e10581/intro.htm

Introduction to Snowflake | Snowflake Documentation. (2023, April 25). https://docs.snowflake.com/en/user-guide-intro

Introduction—Neo4j Graph Data Science. (2023, April 25). Neo4j Graph Data Platform. https://neo4j.com/docs/graph-data-science/2.3/introduction/

ISO/IEC 13249-3:2016(en), Information technology—Database languages—SQL multimedia and application packages—Part 3: Spatial. (n.d.). Retrieved May 10, 2023, from https://www.iso.org/obp/ui/#iso:std:iso-iec:13249:-3:ed-5:v1:en

Jayaram, P. (2018, March 28). Getting started building applications using SQL Server DevOps Tools. *SQL Shack - Articles about Database Auditing, Server Performance, Data Recovery, and More*. https://www.sqlshack.com/getting-started-building-applications-using-sql-server-devops-tools/

JDBC: The Swiss Army Knife of Database Formats. (2017, December 5). Safe Software. https://engage.safe.com/blog/2017/12/jdbc-swiss-army-knife-database-formats/

License Management Services—Oracle Compliance | Oracle. (n.d.). Retrieved May 10, 2023, from https://www.oracle.com/corporate/license-management-services/compliance.html

Logging into Snowflake | Snowflake Documentation. (2023, April 25). https://docs.snowflake.com/en/user-guide/connecting

MashaMSFT. (2023, April 24). *What is Azure SQL? - Azure SQL*. https://learn.microsoft.com/en-us/azure/azure-sql/azure-sql-iaas-vs-paas-what-is-overview

Microsoft. (2023, April 25). *Microsoft Inspire – Join us July 19-20, 2022*. Microsoft Inspire – Join Us July 19-20, 2022. https://inspire.microsoft.com

Microsoft SQL Server Spatial Tools. (2023). [C#]. Microsoft. https://github.com/microsoft/SQLServerSpatialTools (Original work published 2017)

Microsoft SQL Server—Market Share, Competitor Insights in Database. (2023, April 25). https://6sense.com/tech/database/microsoft-sql-server-market-share

MladjoA. (2023a, February 28). *M (Geometry Data Type)—SQL Server*. https://learn.microsoft.com/en-us/sql/t-sql/spatial-geometry/m-geometry-data-type

MladjoA. (2023b, February 28). *Spatial Data Types Overview—SQL Server*. https://learn.microsoft.com/en-us/sql/relational-databases/spatial/spatial-data-types-overview

MladjoA. (2023c, February 28). *Spatial Reference Identifiers (SRIDs)—SQL Server*. https://learn.microsoft.com/en-us/sql/relational-databases/spatial/spatial-reference-identifiers-srids

Monica, P. R. L. (2020, September 16). *Snowflake shares more than double. It's the biggest software IPO ever | CNN Business*. CNN. https://www.cnn.com/2020/09/16/investing/snowflake-ipo/index.html

MSSQLSpatial—Microsoft SQL Server Spatial Database—GDAL documentation. (2023, April 25). https://gdal.org/drivers/vector/mssqlspatial.html

Murray, C., An, N., Betsy George, H. G., & Jayapalan, L. (n.d.). *Topology Data Model Overview* [Topic]. Oracle Help Center; July2022. Retrieved May 10, 2023, from https://docs.oracle.com/en/database/oracle/oracle-database/19/topol/topology-data-model-overview.html#GUID-3F069A32-CCDC-4DBA-87A5-21D501DBBD6D

Murray, C., Anderson, R., Ravada, S., Hu, Y., Abugov, D., Alexander, N., Blackwell, B., Chatterjee, R., Covarrubias, L. A. R., Geringer, D., Horhammer, M., Kazar, B., Kothuri, R., Wang, J., Xie, Q. (Jeffrey), Yang, J., & Jayapalan, L. (n.d.). *Spatial Concepts* [Topic]. Oracle Help Center; April2023. Retrieved May 10, 2023, from https://docs.oracle.com/en/database/oracle/oracle-database/19/spatl/spatial-concepts.html#GUID-67E4037F-C40F-442A-8662-837DD5539784

Murray, C., Anderson, R., Ravada, S., Hu, Y., Abugov, D., Alexander, N., Blackwell, B., Chatterjee, R., Covarrubias, L. A. R., Geringer, D., Horhammer, M., Kazar, B., Kothuri, R., Wang, J., Xie, Q. (Jeffrey), Yang, J., & Jayapalan, L. (2023, April 25). *SDO_PC_PKG Package (Point Clouds)* [Topic]. Oracle Help Center; April2023. https://docs.oracle.com/en/database/oracle/oracle-database/19/spatl/SDO_PC_PKG-reference.html

Murray, C., Fengting Chen, I. L., Zhang, Z., & Jayapalan, L. (2023, April 25). *Preface* [Topic]. Oracle Help Center; May2022. https://docs.oracle.com/en/database/oracle/oracle-database/19/geors/preface.html

Murray, C., Ravada, S., Wang, J., Anderson, R., Hu, Y., Xie, Q. (Jeffrey), Horhammer, M., Abugov, D., Alexander, N., Blackwell, B., Chatterjee, R., Covarrubias, L. A. R., Geringer, D., Kazar, B., Kothuri, R., Yang, J., & Jayapalan, L. (2023, April 25). *Coordinate Systems (Spatial Reference Systems)* [Topic]. Oracle Help Center; January2023. https://docs.oracle.com/en/database/oracle/oracle-database/21/spatl/coordinate-systems-concepts.html#GUID-C3573533-1E3E-42BC-9C43-45AA09EDE150

Neo4j Announces $325 Million Series F Investment, the Largest in Database History. (2023, April 25). Neo4j Graph Data Platform. https://neo4j.com/press-releases/neo4j-announces-seriesf-funding/

Neo4j—Funding, Financials, Valuation & Investors. (2023, April 25). Crunchbase. https://www.crunchbase.com/organization/neo-technology/company_financials

Oracle: A Company (and a Relational Database) in the Making | The Battle of the RDBMSs: SQL Server 2000 Versus Oracle 9i (Part 1) | InformIT. (2023, April 25). https://www.informit.com/articles/article.aspx?p=27317&seqNum=3

Oracle Data Integrator. (2023, April 25). https://www.oracle.com/middleware/technologies/data-integrator.html

Oracle Database. (2023, April 25). https://www.oracle.com/database/

Oracle Graph. (2023, June 1). https://www.oracle.com/database/graph/

Oracle License Audit – Expert Advice. (2022, March 31). https://redresscompliance.com/oracle-license-audit-process-explained-lms-scripts/

Oracle Software Audit Attorneys. (n.d.). Oracle Audits. Retrieved May 10, 2023, from https://oracleaudits.com/

Oracle Spatial—GDAL Documentation. (2023, April 25). https://gdal.org/drivers/vector/oci.html

Overview of the Spark Connector | Snowflake Documentation. (2023, April 25). https://docs.snowflake.com/en/user-guide/spark-connector-overview

PgPointcloud—A PostgreSQL extension for storing point cloud (LIDAR) data. —Pgpointcloud. (2023, April 25). https://pgpointcloud.github.io/pointcloud/

PostGIS — Support. (2023, April 25). https://postgis.net/support/

PostGIS — OSGeo. (2023, April 25). OSGeo. https://www.osgeo.org/projects/postgis/

PostGIS — pgpointcloud. (2023, April 25). https://pgpointcloud.github.io/pointcloud/functions/postgis.html

PostgreSQL / PostGIS — GDAL documentation. (2023, April 25). https://gdal.org/drivers/vector/pg.html

Racine, P. (2021). *PostGIS Workshop* [PLpgSQL]. https://github.com/pedrogit/postgis_workshop (Original work published 2017)

Racine, P. (2023). *PostGIS Add-ons* [PLpgSQL]. https://github.com/pedrogit/postgisaddons (Original work published 2013)

Ramsey, Paul (2022, June 13). Who are the Biggest PostGIS Users? http://blog.cleverelephant.ca/2022/06/postgis-logos.html

Rouault, Even, Warmerdam, Frank, Schwehr, Kurt, Kiselev, Andrey, Butler, Howard, Łoskot, Mateusz, Szekeres, Tamas, Tourigny, Etienne, Landa, Martin, Miara, Idan, Elliston, Ben, Chaitanya, Kumar, Plesea, Lucian, Morissette, Daniel, Jolma, Ari, & Dawson, Nyall. (2023). *GDAL* (v3.7.0). Zenodo. https://doi.org/10.5281/ZENODO.5884351

rwestMSFT. (2023, March 31). *Editions and Supported Features of SQL Server 2022—SQL Server*. https://learn.microsoft.com/en-us/sql/sql-server/editions-and-components-of-sql-server-2022

Safe Software: Home. (2023, April 25). Safe Software. https://www.safe.com/

Simple Feature Access - Part 1: Common Architecture. (2023, April 25). *Open Geospatial Consortium*. https://www.ogc.org/standard/sfa/

Snowflake on Microsoft Azure | Snowflake Cloud Partners. (2023, April 25). Snowflake. https://www.snowflake.com/technology-partners/microsoft/

Snowflake Partner Connect | Snowflake Documentation. (2023, April 25). https://docs.snowflake.com/en/user-guide/ecosystem-partner-connect

Snowflake raises $450 million in another VC round, valued at $3.5 billion. (n.d.). ZDNET. Retrieved May 11, 2023, from https://www.zdnet.com/article/snowflake-raises-450-million-in-another-vc-round-valued-at-3-5-billion/

Spatial and Graph Developer's Guide. (2023, April 25). https://docs.oracle.com/database/121/SPATL/what-is-oracle-spatial-and-graph.htm#SPATL440

Spatial Functions—Cypher Manual. (2023, April 25). Neo4j Graph Data Platform. https://neo4j.com/docs/cypher-manual/5/functions/spatial/

Spatial Values—Cypher Manual. (2023, April 25). Neo4j Graph Data Platform. https://neo4j.com/docs/cypher-manual/current/syntax/spatial

SQL Developer. (2023, April 25). https://www.oracle.com/database/sqldeveloper/

SQL Graph. (2003, June 1). https://learn.microsoft.com/en-us/sql/relational-databases/graphs/sql-graph-overview?view=sql-server-ver16

SQL Server Integration Services—SQL Server Integration Services (SSIS). (2023b, March 3). https://learn.microsoft.com/en-us/sql/integration-services/sql-server-integration-services

Surveyor dies after being hit by pickup truck in Lakeland. (2020, February 8). Wtsp.Com. https://www.wtsp.com/article/news/local/polkcounty/lakeland-crash-kills-surveyor/67-aca4c7a8-f279-4c85-9c49-153447d14961

The PostGIS Open Source Project on Open Hub. (2023, April 25). https://www.openhub.net/p/postgis

Topological Sort—Neo4j Graph Data Science. (2023, April 25). Neo4j Graph Data Platform. https://neo4j.com/docs/graph-data-science/2.3/algorithms/alpha/topological-sort/

Unsupported Data Types | Snowflake Documentation. (2023, April 25). https://docs.snowflake.com/en/sql-reference/data-types-unsupported

Using FME and the Snowflake Database Format. (2023, April 25). https://community.safe.com/s/article/catching-snowflakes-with-fme-introducing-fme-and-t

Using Snowflake Spatial in FME for Geospatial Data. (n.d.). Retrieved May 10, 2023, from https://community.safe.com/s/article/using-snowflake-spatial-in-fme-for-geospatial-data

What is IFC(Industry Foundation Class) in BIM? (2023, April 25). https://www.midasbridge.com/en/blog/importance-of-ifc-in-bim

What's New in ArcGIS Knowledge. (2023, April 25). *ArcGIS Blog*. https://www.esri.com/arcgis-blog/products/arcgis-knowledge/announcements/whats-new-in-arcgis-knowledge/

WilliamDAssafMSFT. (2023, March 3). *What is the Azure SQL Database Service? - Azure SQL Database*. https://learn.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview

Woman Gets 10 Years For Killing Surveyor With Car. (2023, April 25). https://www.news4jax.com/news/2009/01/22/woman-gets-10-years-for-killing-surveyor-with-car/