PostgreSQL / PostGIS Warehouses

Arbeiten mit PostgreSQL / PostGIS in GeoMedia und GEOS Pro

Version vom 07.07.2026

Hexagon Technology Center GmbH
Octave | Niederlassung Zürich
Flurstrasse 55
8048 Zürich
Schweiz
+41 (0)43 322 46 46
info.switzerland.sig@octave.com
www.octave.com



Inhaltverzeichnis

1 Einleitung ... 3

2 Voraussetzungen ... 3

2.1 Datenbank ... 3

2.2 GeoMedia ... 3

2.3 GEOS Pro ... 3

3 Arbeiten mit GeoMedia ... 4

3.1 Voraussetzungen und Grundsätze ... 4

3.2 Erzeugen der GeoMedia Metadaten ... 4

3.3 Verbindung in GeoMedia erstellen ... 5

3.4 Arbeiten mit unterschiedlichen Schemas ... 6

3.4.1 Erzeugen neuer Features ... 6

3.4.2 Auffinden neuer Features ... 6

3.5 Datenhaltung in PostGIS ... 6

3.5.1 Datentypenmapping ... 6

3.5.2 Geometriespeicherung ... 7

3.5.3 Tabellennamen ... 8

4 Arbeiten mit GEOS Pro ... 8

4.1 Organisation der *GEOS Pro* Metadaten ... 8

4.2 Datenbank vorbereiten ... 8

4.2.1 Allgemeine Voraussetzungen ... 8

4.2.2 Verwendung der Variable *search_path* ... 8

4.3 Konvertierung von Operaten nach *PostGIS* ... 8

4.3.1 Kopieren der Datenbank nach *PostGIS* ... 9

4.3.2 Umhängen des Geworkspaces ... 9

4.4 Arbeiten mit unterschiedlichen Schemas ... 10

5 Tipps und Tricks ... 11

5.1 Passwort nicht speichern ... 11

5.2 *Search_path* wird nicht nachgeführt ... 11

5.3 GRIVIS GEOS Suchen über SQL definieren ... 11

1 Einleitung

Seit GeoMedia 2016 ist der Datenserver für PostgreSQL / PostGIS fester Bestandteil von GeoMedia. PostgreSQL ist eine objektrelationale Datenbank, welche als Open-Source Software kostenfrei verfügbar ist. PostGIS ist eine Erweiterung für PostgreSQL und ermöglicht die Verwaltung und Auswertung von räumlichen Daten.

Dieses Dokument beschreibt verschiedene Aspekte von PostgreSQL / PostGIS in Kombination mit GeoMedia und GEOS Pro. In den weiteren Kapiteln wird die Datenbank kurz PostGIS genannt, analog der Benennung der Warehouse-Verbindung in GeoMedia.

2 Voraussetzungen

2.1 Datenbank

Für die Verwendung der Datenbank in GeoMedia und GEOS Pro ist eine Installation von PostgreSQL mit Erweiterung PostGIS nötig. Die empfohlenen Datenbankversionen entnehmen Sie bitte den jeweiligen Release Notes der verwendeten GeoMedia-Version.

2.2 GeoMedia

Für die Verbindung mit einer PostgreSQL / PostGIS-Datenbank ist keine Installation von zusätzlicher Client-Software nötig (im Gegensatz zu Oracle). Der Verbindungstyp wird installiert, unabhängig davon, ob eine PostGIS Installation vorhanden ist. Eine Warehouse-Verbindung wird über den Servernamen und Datenbank erstellt. Eine Verbindung kann nur auf existierende Datenbanken hergestellt werden.

Der Datenbankserver von GeoMedia existiert schon seit längerer Zeit. Die vorliegende Dokumentation bezieht sich auf die Gegebenheiten in GeoMedia 2020 Update 2. Für die Arbeit mit PostgreSQL / PostGIS. Für die Arbeit mit PostGIS ist stehts auf eine möglichst aktuelle GeoMedia-Version zu achten, da der Datenserver noch grossen Änderungen und Verbesserungen unterläuft.

2.3 GEOS Pro

PostGIS Datenbanken werden seit GEOS Pro 3.10 vollständig unterstützt. Mit der Version 3.11 können auch Daten ausserhalb des «public» Schemas mit GEOS Pro bearbeitet werden. Die Metadatentabellen von GEOS Pro werden in jedem Schema mit GEOS Pro Daten separat geführt.

3 Arbeiten mit GeoMedia

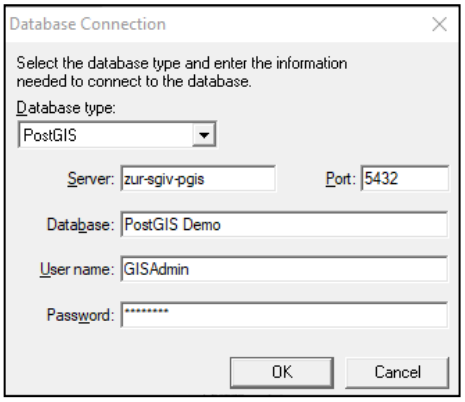
3.1 Voraussetzungen und Grundsätze

Folgende Bedingungen muss eine Datenbank für eine erfolgreiche Verbindung erfüllen:

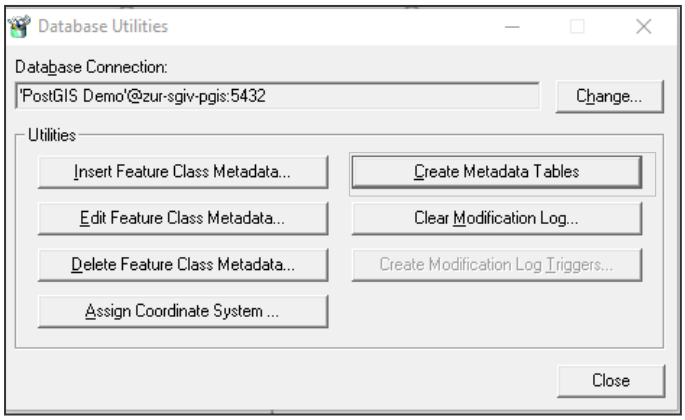
- Extension PostGIS muss für diese Datenbank erzeugt worden sein
- Namen von Tabellen, Views und Feldern müssen in Kleinbuchstaben ausgedrückt sein (Objekte mit Grossbuchstaben im Namen werden vom Datenserver unterdrückt)
- Datenbanken ohne GeoMedia-Metadatentabellen sind Read-Only, für eine Read-Write-Verbindung ist die Erstellung der Metadatentabellen vorausgesetzt
- Für einen Read-Write-Zugriff auf einzelne Tabellen müssen die entsprechenden Einträge in den Metadatentabellen vorhanden sein

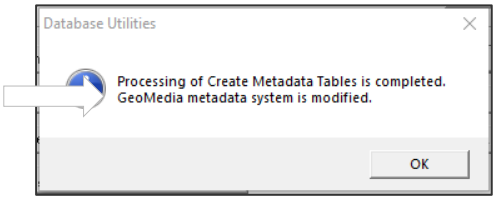
3.2 Erzeugen der GeoMedia Metadaten

Hierfür nutzen Sie das Datenbankdienstprogramm von GeoMedia. Nach Auswahl von „PostGIS“ als Datenbanktyp können Sie die Verbindungsparameter eingeben:

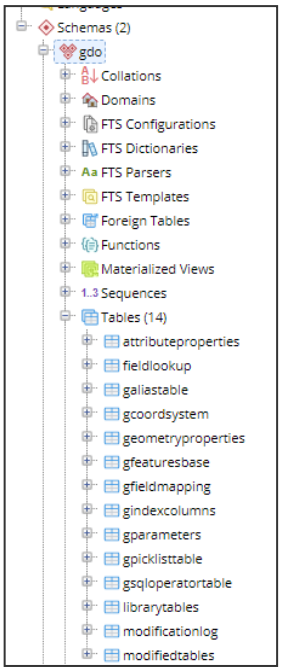


Über den Befehl Create Metadata Tables werden die Metadatentabellen für GeoMedia erzeugt:



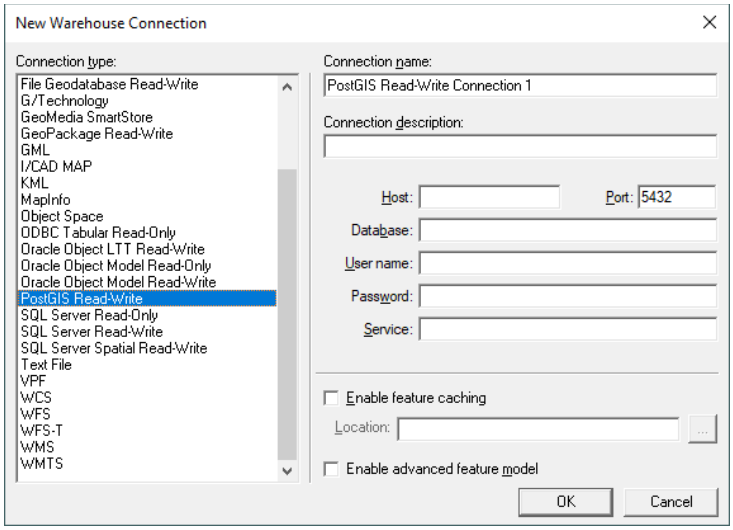


Hierdurch wird in der Datenbank ein Schema gdo angelegt, welches die üblichen Metadatentabellen von GeoMedia enthält:



3.3 Verbindung in GeoMedia erstellen

In GeoMedia können PostGIS-Datenbanken über den Verbindungstyp PostGIS erstellt werden:



Eine Verbindung kann über die folgenden Varianten definiert werden

- Kombination aus Host / Port / Datenbank / User / Passwort
- Servicename gemäss pg_service.conf

Die Kombination aus User und Datenbank muss pro Verbindung in GeoMedia eindeutig sein. Es können nicht mehrere Verbindungen auf die gleiche Datenbank mit dem gleichen User erstellt werden.

3.4 Arbeiten mit unterschiedlichen Schemas

Seit GeoMedia 2018 Update 2 wird auch das Verwalten und Verwenden von Daten in anderen Schemen als public ermöglicht. Folgende Regeln gelten bei der Verwendung von Schemen in GeoMedia.

- Objekte aus einem anderen Schema als public werden in der Form <schema name>.<feature class name> dargestellt
- Objekte aus dem public Schema werden in der einfachen Form <feature class name> dargestellt

3.4.1 Erzeugen neuer Features

Das Erzeugen neuer Features folgt folgenden Grundsätzen:

- Wenn ein Feature mit Schema-Präfix erzeugt wird, dann wird das unter dem genannten Schema erzeugt.
- Wenn ein Feature ohne Schema-Qualifikation erstellt wird, wird es unter dem durch den PostgreSQL-Suchpfad festgelegten Standardschema (erstes Schema) erstellt. In der Standardkonfiguration ist dies "public".

3.4.2 Auffinden neuer Features

Beim Auffinden von Features kommen folgende Grundsätze in GeoMedia zu tragen:

- Wenn ein Objekt in einer Select- oder einer anderen Operation mit einer Schema-Qualifikation referenziert wird, wird das Schema berücksichtigt und die Tabelle wird so gefunden.
- Wenn ein Objekt in einer Select- oder einer anderen Operation ohne Schema-Qualifikation referenziert wird, wird angenommen, dass sich das Objekt im «public» Schema befindet.

3.5 Datenhaltung in PostGIS

3.5.1 Datentypenmapping

Die folgende Tabelle listet die Zuordnung von Datentypen aus PostGIS in GeoMedia.

GeoMedia Datentyp	PostGIS Datentyp
Boolean	boolean
Byte	"char"
Integer	smallint
Long	Integer, serial for AutoNumber
Single	real
Double	Double precision
Currency	money

Date	Timestamp without time zone
Text	Character varying(n)
LongBinary	bytea
Memo	text
Guid	uuid
Spatial	geometry
Graphic	geometry

3.5.2 Geometriespeicherung

Der PostGIS-Datenserver verwendet zwei Spalten für das Speichern von Geometrie-informationen:

Name	**Datentyp**	**Beschreibung**
geometry_spa	geometry	Standard-*PostGIS*-Geometrie-Informationen
geometry	bytea	*GeoMedia*-spezifischen Informationen in einem Binärformat. (Hier werden von *PostGIS* nicht unterstützte Geometrietypen wie Punkt mit Orientierung (Drehwinkel), Text und Raster(-Einpassung) gespeichert.)

Die Benennung der beiden Spalten ist grundsätzlich frei. Die beiden Spaltennamen müssen einfach entsprechend in den GeoMedia-Metadatentabellen abgelegt werden. Die oben abgebildete Tabelle zeigt die von GeoMedia standardmässig verwendete Namenskonvention.

Mit GeoMedia 2020 ist die Binärspalte (GDO-Spalte) gemäss dieser Tabelle für die Geometriespeicherung nötig:

***GeoMedia* Geometrietyp**	**Keine GDO-Spalte**	**GDO-Spalte Optional**	**GDO-Spalte obligatorisch**
Fläche	**x**		
Linie	**x**		
Punkt		**x**	
Zusammengesetzt (Compound)		**x**	
Raster			**x**
Text			**x**

Weitere Informationen zur Abbildung der unterschiedlichen Geometrietypen aus GeoMedia in PostGIS-Geometrien können der Hilfe von GeoMedia entnommen werden.

3.5.3 Tabellennamen

PostgreSQL verwendet für Tabellennamen grundsätzlich Kleinbuchstaben, der Import-Befehl berücksichtigt dies automatisch und passt die Tabellennamen automatisch an. So wird beispielsweise aus dem *Shape*-Dateinamen 'Veloweg' automatisch 'veloweg'.

Nach *ANSI*-Standard *SQL*-Regeln besteht ein Tabellenname ausschließlich aus Buchstaben, mit Ausnahme des Unterstrichs (_); nur solche Tabellennamen sind für die *GeoMedia-GDO*-Datenbank-Schnittstelle kompatibel. Tabellennamen mit einem Minus-Zeichen (-) sind beispielsweise ungültig.

4 Arbeiten mit GEOS Pro

4.1 Organisation der *GEOS Pro* Metadaten

Die Metadaten von *GEOS Pro* werden in *PostGIS* pro Schema separat angelegt, wohingegen die Metadaten für *GeoMedia* zentral pro Datenbank im Schema *gdo* gehalten werden. Diese Aufteilung ist mit derjenigen in Oracle Datenbanken zu vergleichen. Somit kann jedes Schema als eigenes *GEOS Pro* Operat verstanden werden.

4.2 Datenbank vorbereiten

4.2.1 Allgemeine Voraussetzungen

Für das Arbeiten von *GEOS Pro* mit *PostGIS*-Datenbanken gelten die gleichen Voraussetzungen wie für *GeoMedia* (siehe Kapitel Arbeiten mit *GeoMedia* > Voraussetzungen und Grundsätze). Da *GEOS Pro* auch Schreibend auf Datenbanken zugreift ist das *GeoMedia*-Metadatenschema *gdo* ebenfalls zwingend.

4.2.2 Verwendung der Variable *search_path*

Für das Auffinden von Tabellen und schreiben von Daten extrahiert *GEOS Pro* das erste Schema aus der Variable *search_path* mit Hilfe der *PostgreSQL*- Funktion *current_schema()* extrahiert. Für die Datenhaltung in anderen Schemen als *Public* finden sich im Kapitel 'Arbeiten mit *GEOS Pro* > Arbeiten mit unterschiedlichen Schemas' weitere Informationen.

In *GEOS Pro* 3.11 erscheint im INERLIS Import/Export-Dialog ein Popup-Fenster, welches das aktuelle Schema gemäss der Funktion *current_schema()* anzeigt.



Achtung: Die Variable *search_path* kann auf unterschiedlichen Stufen (Server, Datenbank, User, Session...) definiert werden.

4.3 Konvertierung von Operaten nach *PostGIS*

Für die Konvertierung von bestehenden Operaten nach *PostGIS* stellt *GEOS Pro* Werkzeuge zur Verfügung, um die Datenbank und Geoworkspaces zu konvertieren.

4.3.1 Kopieren der Datenbank nach *PostGIS*

1. Leere Datenbank erstellen
    a. Leere Datenbank erstellen
    b. *PostGIS* Extension erzeugen
    c. Metadatentabellen mit Datenbankdienstprogramm erzeugen
    d. Optional: Schema erzeugen und *Search_path* konfigurieren

2. *GeoMedia* öffnen (z.B. Geoworkspacevorlage verwenden)
    a. Verbindung auf bestehendes Operat erstellen
    b. Verbindung auf leere *PostGIS* Datenbank aus Schritt 1 erstellen

3. *GEOS Pro* > INTERLIS Import/Export öffnen
    a. Quelldatenbank auswählen
    b. Verbindung > Datenbank öffnen
    c. Zieldatenbank (*PostGIS*) wählen und starten

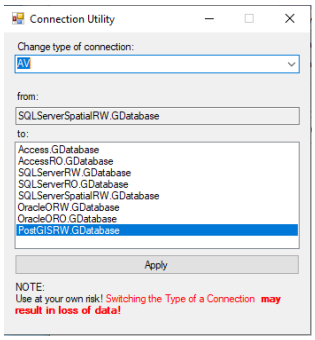
4. Definitionen AV neu speichern im Zieloperat (*PostGIS*)
    a. *GEOS Pro* > Administrator > Einstellungen > Definitionen AV öffnen
    b. Einstellungen mit 'Werte übernehmen' erneut abspeichern
    ➔ Dieser Schritt ist nur für Operate der amtlichen Vermessung nötig

4.3.2 Umhängen des Geworkspaces

Das Dienstprogramm ConnectionUtility.exe kann bei Octave auf Anfrage kostenlos bezogen werden.

1. Umzuhängenden Geworkspace (*.gws/*.gwt) öffnen (Achtung: nur 1 Geworkspace darf geöffnet werden)

2. ConnectionUtility.exe starten
    a. Zu Ändernde Verbindung im Dropdown auswählen
    b. Gewünschter Verbindungstyp (hier *PostGIS*) auswählen
    c. Änderung anwenden

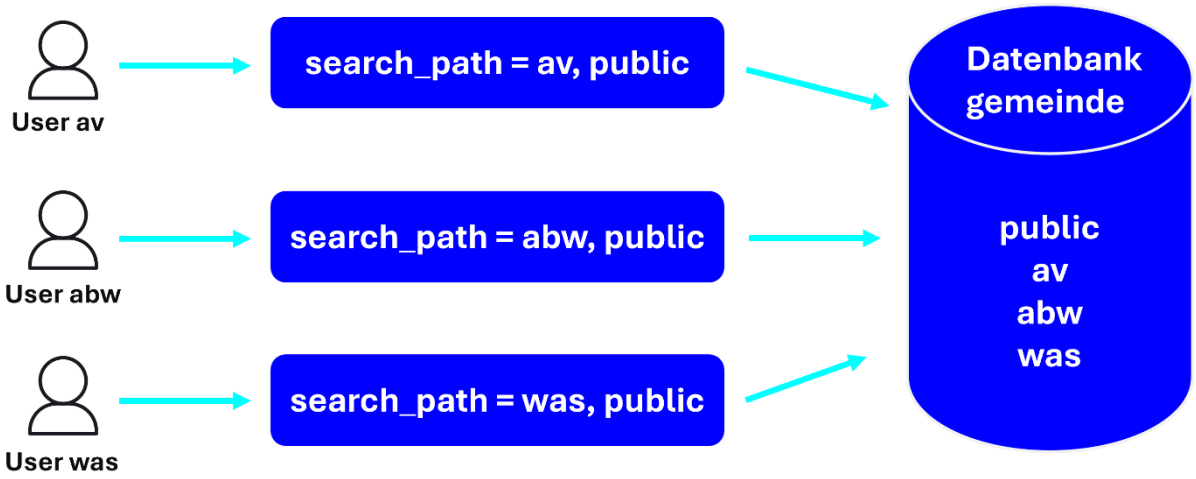


3. Zurück zu *GeoMedia* wechseln
    a. Geoworkspace neu speichern
    b. Geoworkspace schliessen
    c. Geoworkspace neu öffnen
    d. Verbindungseigenschaften der geänderten Verbindung anpassen

4.4 Arbeiten mit unterschiedlichen Schemas

GEOS Pro verwendet für die Lese- und Schreibzugriffe das erste Schema im *search_path*. Im Gegensatz zu *PostgreSQL* an sich, werden weitere Schemas im *search_path* nicht berücksichtigt für das Auffinden von Tabellen. Für die Extraktion eines korrekten Schemas wird die *PostgreSQL*-Funktion *current_schema()* verwendet, diese führt eine vorgängige Prüfung der Einträge durch. Eine weitere Bedingung ist, dass die Kombination aus Datenbank und User für eine Verbindung in *GeoMedia* eindeutig sein muss.

Eine mögliche Architektur mit *GEOS Pro* könnte somit wie folgt aussehen:



Folgende Voraussetzungen müssen für die Verwendung von anderen Schemas als *public* erfüllt sein:

- Das Schema, in welchem die Daten gehalten werden, muss an erster Stelle im *search_path* geführt werden und somit als Rückgabewert der Funktion *current_schema()* dienen
- Das Schema, in welchem die Funktionen der Extension *PostGIS* verwaltet werden (standardmässig '*public*) muss ebenfalls im *search_path* hinten mitgeführt werden, da sich in diesem Schema für die Erzeugung der Tabellen nötige Funktionen verbergen.
- Der für die Verbindungserstellung verwendete User muss auf dem gewünschten Schema autorisiert werden und über alle notwendigen Rechte verfügen (CREATE SCHEMA av AUTHORIZATION av;)
- Metadatenschema *gdo* von *GeoMedia* muss vorhanden sein

In *GEOS Pro* 3.11 erscheint im INERLIS Import/Export-Dialog ein Popup-Fenster, welches den ersten Eintrag der Variable *search_path* als Information ausgibt.

Achtung: Die Variable *search_path* kann auf unterschiedlichen Stufen (Server, Datenbank, User, Session...) definiert werden.

5 Tipps und Tricks

5.1 Passwort nicht speichern

GeoMedia speichert standardmässig das Passwort für *PostGIS*-Verbindungen im Geoworkspace. Wenn diese Funktionalität nicht gewünscht ist, dann kann dies über den folgenden Registry-Eintrag geändert werden:

```
HKEY_LOCAL_MACHINE\\SOFTWARE\\WOW6432Node\\GDO\\PostGIS Read-Write\\1.0\\Store Password
```



Standardmässig ist dieser Wert auf 1 gesetzt (Passwörter werden gespeichert). Soll der User zur Passworteingabe gezwungen werden, dann kann der Wert auf 0 gesetzt werden.

5.2 *Search_path* wird nicht nachgeführt

Wird über eine Datenbankabfrage oder über *GeoMedia* / *GEOS Pro* ein 'falscher' *search_path* angezeigt, kann dies verschiedene Gründe haben:

- Wenn der *search_path* auf Ebene Datenbank oder Server gesetzt wird, muss eine neue Verbindung geöffnet werden, damit die Einstellungen wirken
- Der *search_path* auf Stufe User unterscheidet sich vom *search_path* auf Stufe Datenbank. Beim Ändern der Variable *search_path* muss genau beachtet werden, auf welchem Niveau diese geändert wird und in welcher Reihenfolge diese von *PostgreSQL* berücksichtigt wird:
    - Funktion – behält ihre Gültigkeit während der Ausführung einer Funktion: ALTER FUNCTION some_func() SET *search_path*=xy, *public*;
    - Session – behält ihre Gültigkeit während einer Session: set *search_path*= xy, *public*;
    - Datenbankuser – wirkt erst nachdem die Verbindung neu aufgebaut wurde: ALTER ROLE postgres IN DATABASE mydb SET *search_path* = xy, *public*;
    - Server – wirkt erst nachdem die Verbindung neu aufgebaut wurde: ALTER ROLE postgres SET *search_path* = xy, *public*;
    - Datenbank – wirkt erst nachdem die Verbindung neu aufgebaut wurde: ALTER DATABASE mydb SET *search_path* = xy, *public*;
    - Die *search_path* Variablen werden in der oben aufgeführten Reihenfolge voneinander übersteuert.
- Der *search_path* wurde versehentlich nur für die aktuelle Session geändert und nicht permanent
- Der *search_path* der Datenbank wurde nicht gesetzt

5.3 GRIVIS GEOS Suchen über SQL definieren

Beim Einrichten von Suchabfragen über SQL ist das Schema innerhalb der Bedingung mitzuführen. Der Tabellenname für das Suchresultat kann ohne Schema verwendet werden:

```
a10_lie_liegenschaft WHERE liegenschaft_von IN (SELECT gid FROM meinSchema.a10_lie_grundstueck WHERE nummer='?')
```

Über Octave

Octave bietet geschäftskritische Software, die Unternehmen befähigt, in jeder Phase des Asset-Lebenszyklus – bei Design, Entwicklung, Betrieb und Schutz – fundierte Entscheidungen zu treffen. Dies gilt besonders, wo Leistung, Sicherheit und Zuverlässigkeit unverzichtbar sind und Misserfolg keine Option ist. Durch die Umwandlung komplexer operativer Daten in nutzbare Mehrwerte verbindet Octave tiefgreifende Expertise, reale Bedingungen und unternehmensweite Einblicke, um Leistung, Resilienz und Reaktionsfähigkeit zu verbessern.

Im Jahr 2025 leitete Hexagon AB eine Ausgliederung seiner Divisionen Asset Lifecycle Intelligence sowie Safety, Infrastructure & Geospatial ein. Zusammen mit ETQ und Bricsys entstand aus dieser Ausgliederung Octave, das künftig als eigenständiges Unternehmen agieren wird. Octave stützt sich dabei weiterhin auf fundiertes Fachwissen, langjährige Erfahrung und Beziehungen, die über Jahrzehnte hinweg aufgebaut wurden.

Octave beschäftigt mehr als 7.000 Mitarbeiter in 45 Ländern. Erfahren Sie mehr unter **octave.com** and folgen Sie uns auf **LinkedIn**.

©2026 Intergraph Corporation und/oder ihre verbundenen Unternehmen. Alle Rechte vorbehalten. Das Octave-Logo ist eine Marke der Intergraph Corporation. Andere Marken und Produktnamen sind registrierte Warenzeichen der jeweiligen Eigentümer.

Intergraph Corporation
305 Intergraph Way
Madison, AL 35758
USA
T: +1 256 730 2000
www.octave.com

Hexagon Technology Center GmbH
Flurstrasse 55
8048 Zürich
Schweiz
T: +41 43 322 46 46
www.octave.com