



ISO 19125 Simple Feature Access



GEO2311 H2015 – Sverre Stikbakke

ISO 19125 – Simple feature access

▶ Part 1: Common architecture

- ▶ Geometry object model
- ▶ Well-known Text Representation for Geometry
- ▶ Well-known Binary Representation for Geometry
- ▶ Well-known Text Representations of Spatial Reference Systems

▶ Part 2: SQL option: Tre ulike db-skjema

- ▶ Normalized Geometry Schema
- ▶ Binary Geometry Schema
- ▶ Geometry Type Schema

To standarder med ulikt fokus

- ▶ ISO 19107 Spatial Schema:
 - ▶ “Spatial schema is an abstract and non-platform dependent specification”
- ▶ ISO 19125 Simple Features Access -- Part 1: Common architecture:
 - ▶ “SFA-CA is an implementation and platform dependent specification”
 - ▶ Opprinnelig utviklet av OGC

Simple Features (officially **Simple Feature Access**) is both an [Open Geospatial Consortium](#) (OGC) and [International Organization for Standardization](#) (ISO) standard **ISO 19125** that specifies a common storage and access model of mostly [two-dimensional geographical data](#) (point, line, polygon, multi-point, multi-line, etc.)

The ISO 19125 standard comes in two parts. Part one, ISO 19125-1 (SFA-CA for "common architecture"), defines a model for two-dimensional simple features, with linear interpolation between vertices.

The data model defined in SFA-CA is a hierarchy of [classes](#).

This part also defines representation using [Well-Known Text](#) (and Binary).

Part 2 of the standard, ISO 19125-2 (SFA-SQL), defines an implementation using [SQL](#).^[1]

https://en.wikipedia.org/wiki/Simple_Features



Litteratur

- ▶ OGC-spesifikasjonene er fritt tilgjengelige:
- ▶ Se www.opengeospatial.org/docs/is:
 - ▶ [OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 1: Common architecture](#)
 - ▶ [OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 2: SQL option](#)

Bygger på

- ▶ ISO/IEC CD 13249-3:2006(E) – Text for FDIS Ballot
Information technology – Database languages
– SQL Multimedia and Application Packages
— Part 3: Spatial, May 15, 2006.
- ▶ ISO 19107, Geographic information – Spatial schema
- ▶ ISO 19111, Geographic information – Spatial referencing
by coordinates
- ▶ ISO 19133, Geographic information – Location based
services – Tracking and navigation
- ▶ Se ISO/TC211 [Standards Guide](#)

Dimensjoner

x

y

z

m – for measurement

Alle operasjoner foregår i kartplanet.

Det vil si at z og m blir tatt vare på, men inngår ikke i beregninger.
(se 6.1.2.5)



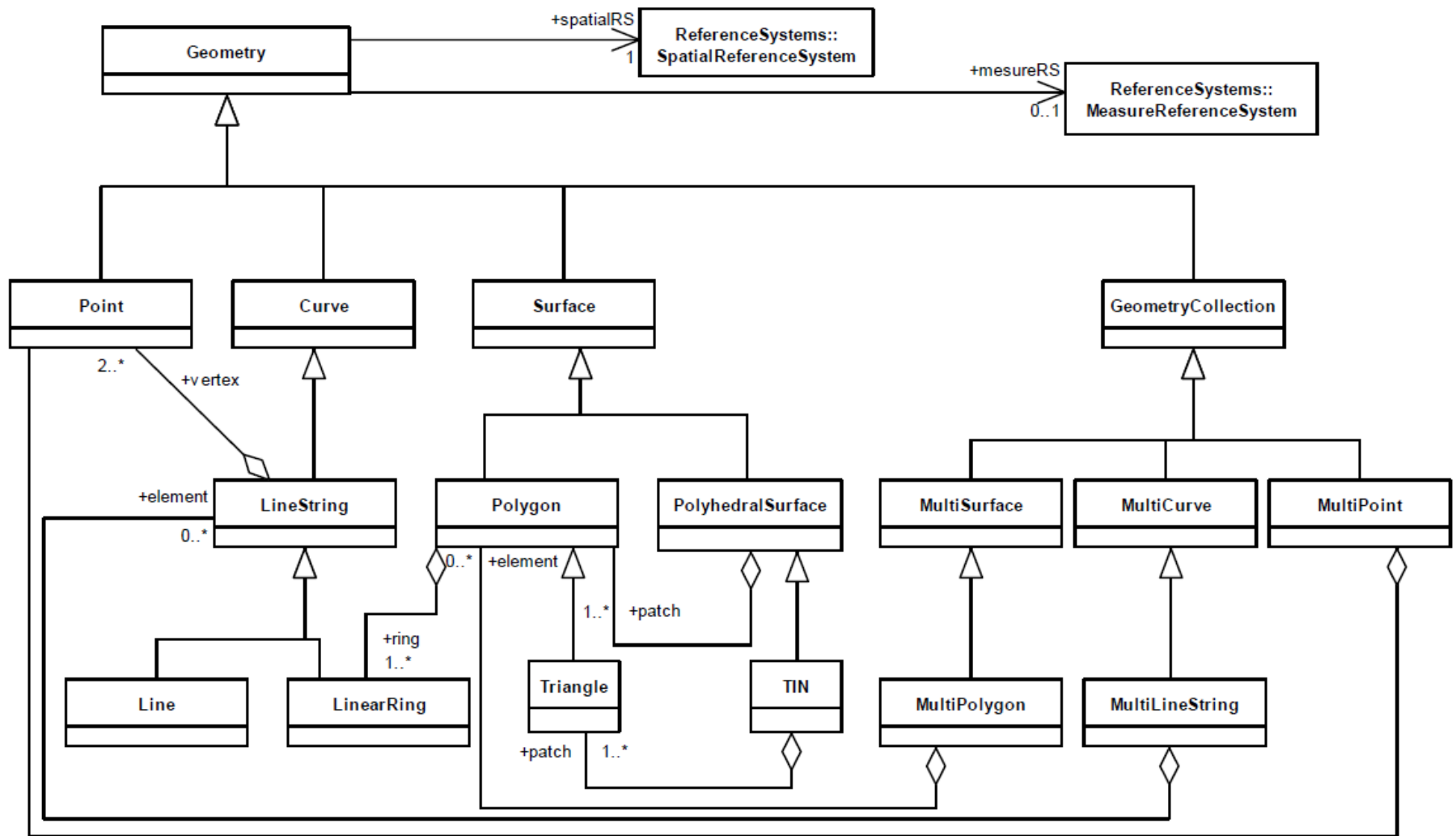
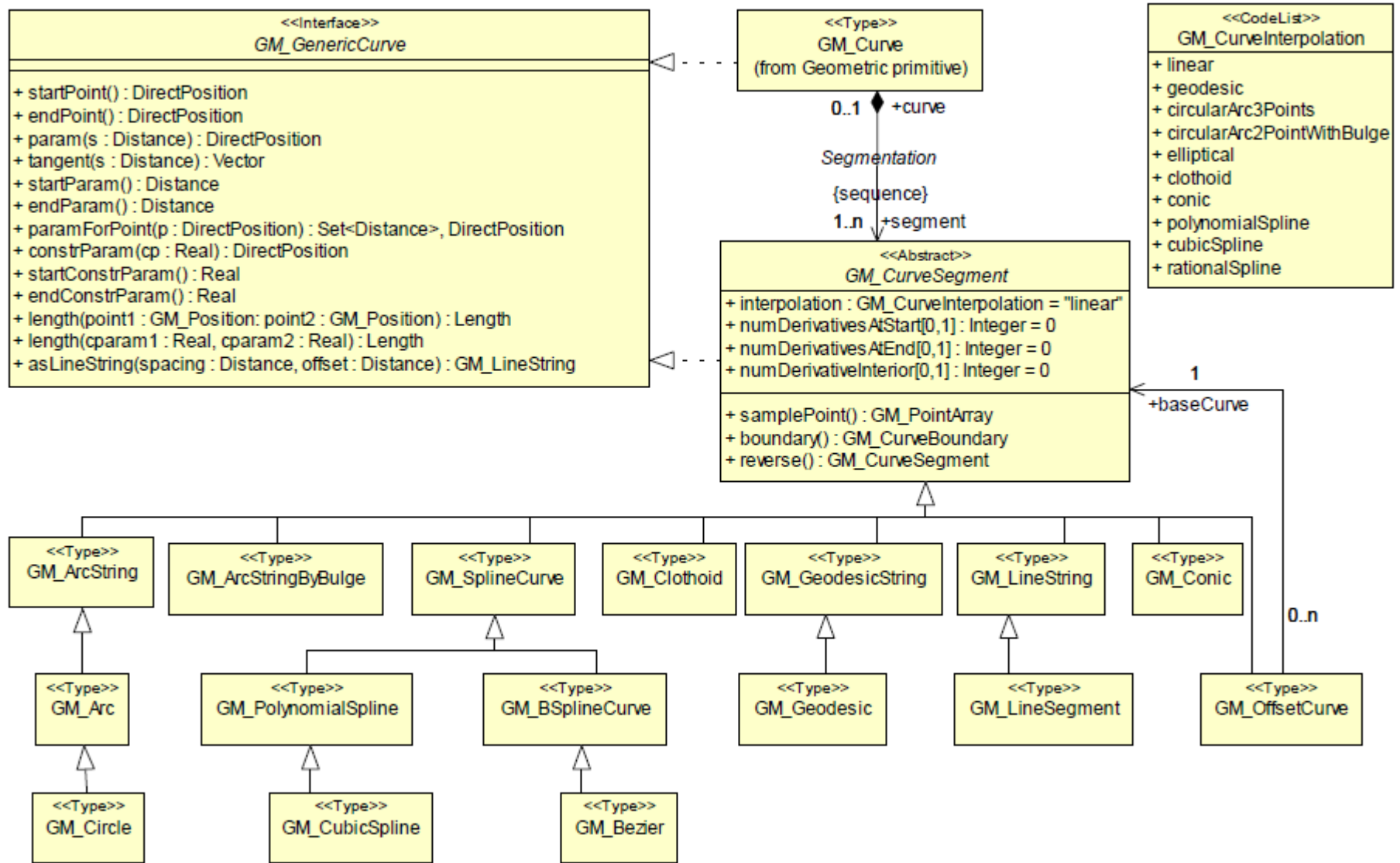


Fig. I: Geometry object model



Linestring er en spesialisering av Curve-klassen med lineær interpolering mellom punktene som inngår (6.1.7.1).

Dette tilsvarer KURVE i SOSI.

Denne standarden har ikke noe som tilsvarer SOSI-typene

BUEP – bue oppgitt med tre koordinatpar

SIRKELP – sirkel oppgitt med tre koordinatpar

KLOTOIDE – overgangskurve angitt som del av en spiralbue

BEZIER

```
.BUEP I:  
..OBJTYPE GangSykkelveg  
..NØ  
101000 602000  
107500 601500  
100500 600000
```



Metoder på alle geomtriobjekter

- ▶ Dimension()
- ▶ GeometryType()
- ▶ SRID() – Spatial Reference System ID
- ▶ Envelope() – minimum bounding box
- ▶ AsText()
- ▶ AsBinary()
- ▶ IsEmpty()
- ▶ IsSimple()
- ▶ Is3D()
- ▶ IsMeasured()

Metoder for å teste geografisk beliggenhet

- ▶ Equals()
- ▶ Disjoint()
- ▶ Intersects()
- ▶ Touches()
- ▶ Crosses()
- ▶ Within()
- ▶ Contains()
- ▶ Overlaps()
- ▶ ++

Metoder for geografisk analyse

- ▶ Distance(anotherGeometry: Geometry)
- ▶ Buffer()
- ▶ ConvexHull()
- ▶ Intersection()
- ▶ Union()
- ▶ Difference()
- ▶ SymDifference()

Metoder spesifikt for den enkelte geometritype

▶ Eks. Point:

- ▶ `x()`
- ▶ `y()`

▶ Eks. Curve:

- ▶ `Length()`
- ▶ `IsClosed()`

▶ Eks. Surface:

- ▶ `Area()`

Geometry collections

- ▶ Kan referere til en samling av geometriske objekter
 - ▶ Point - Multipoint
 - ▶ Linestring – Multilinestring
 - ▶ Polygon - Multipolygon

PolyhedralSurface

- ▶ **TIN – Triangular Irregular Network**
 - ▶ Nettverk av trekanter
 - ▶ Brukes ofte til å representere terrengoverflate (3D)
- ▶ **PolyhedralSurface**
 - ▶ Sammenhengende overflate av polygoner
 - ▶ Kan brukes til å modellere objekter i 3D

Table 6: Example Well-known Text Representation of Geometry

Geometry Type	Text Literal Representation	Comment
Point	Point (10 10)	a Point
LineString	LineString (10 10, 20 20, 30 40)	a LineString with 3 points
Polygon	Polygon ((10 10, 10 20, 20 20, 20 15, 10 10))	a Polygon with 1 exteriorRing and 0 interiorRings
Multipoint	MultiPoint ((10 10), (20 20))	a MultiPoint with 2 points
MultiLineString	MultiLineString ((10 10, 20 20), (15 15, 30 15))	a MultiLineString with 2 linestrings
MultiPolygon	MultiPolygon (((10 10, 10 20, 20 20, 20 15, 10 10)), ((60 60, 70 70, 80 60, 60 60)))	a MultiPolygon with 2 polygons

Se flere eksempler under 7.2.6, eventuelt en.wikipedia.org/wiki/Well-known_text



Well-known Binary Representation for Geometry

```
// Basic Type definitions
// byte : 1 byte
// uint32 : 32 bit unsigned integer (4 bytes)
// double : double precision number (8 bytes)

// Building Blocks : Coordinate, LinearRing
Point {
    double x;
    double y}

PointZ {
    double x;
    double y;
    double z}
```

Dette er eksempel på programkode i programmeringsspråket C++.
Flere eksempler under 8.2.8.



Well-known Text Representation of Spatial Reference Systems

- ▶ Geografisk (, breddegrad, lengdegrad)
- ▶ Projisert (x, y)
- ▶ Geosentrisk (x, y, z) (brukes bl.a. ved GNSS-målinger)

```
PROJCS["NAD_1983_UTM_Zone_10N",  
GEOGCS["GCS_North_American_1983",  
DATUM[  
"D_North_American_1983", ELLIPSOID["GRS_1980", 6378137, 298.257222101]],  
PRIMEM["Greenwich", 0], UNIT["Degree", 0.0174532925199433]],  
PROJECTION["Transverse_Mercator"], PARAMETER["False_Easting", 500000.0],  
PARAMETER["False_Northing", 0.0], PARAMETER["Central_Meridian", -123.0],  
PARAMETER["Scale_Factor", 0.9996], PARAMETER["Latitude_of_Origin", 0.0],  
UNIT["Meter", 1.0]]
```

Simple Feature access- Part 2: SQL Option (ISO 19125 - 2)

► Bygger på

- ISO 19125 og det som den bygger på
- I tillegg:
 - ISO/IEC 9075-1, Information technology — Database languages — SQL — Part 1 – Part 5
 - ISO 19109, Geographic information — Rules for application schema
 - ISO 19119, Geographic information — Services

Simple features: Tre ulike db-skjema

- ▶ Normalized Geometry Schema
- ▶ Binary Geometry Schema
- ▶ Geometry Type Schema

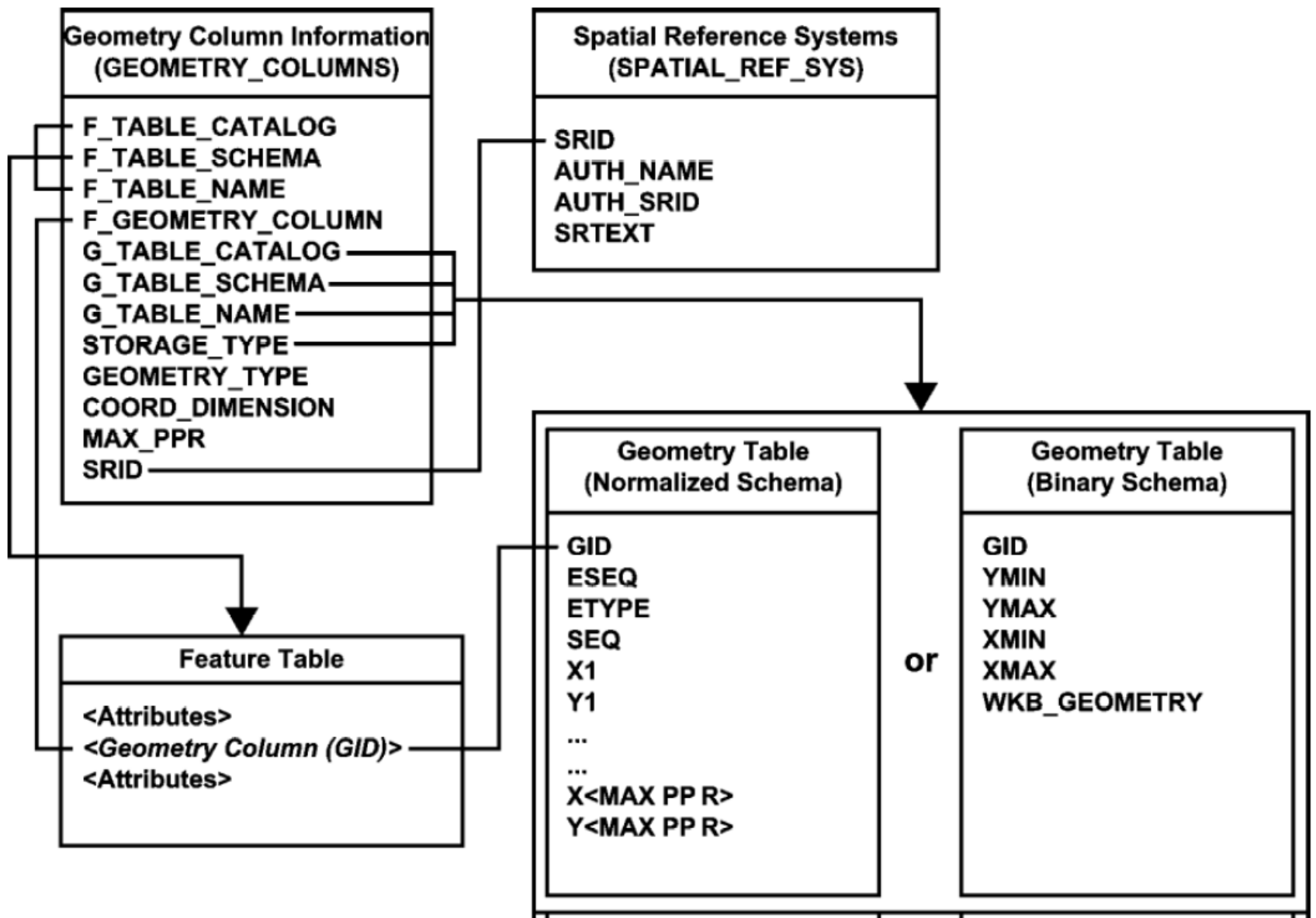
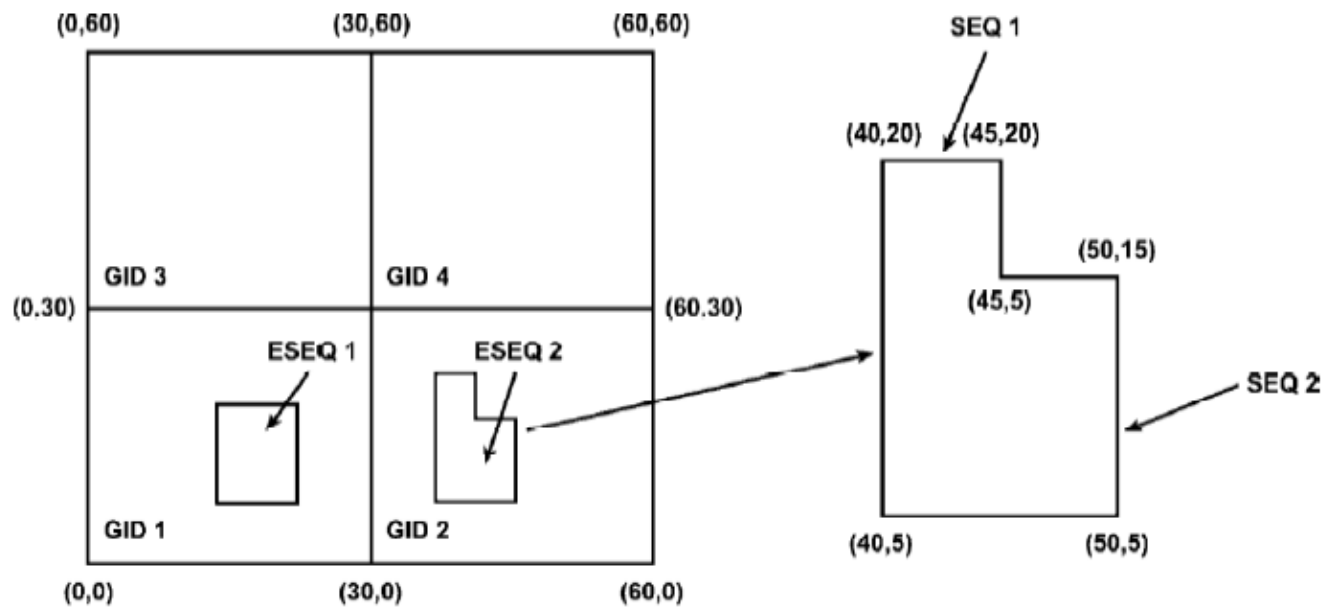


Figure 1: Schema for feature tables using predefined data types



GID 1	ESEQ	ETYPE	SEQ	X0	Y0	X1	Y1	X2	Y2	X3	Y3	X4	Y4
1	1	3	1	0	0	0	30	30	30	30	0	0	0
1	2	3	1	10	10	10	20	20	20	20	10	10	10
2	1	3	1	30	0	30	30	60	30	60	0	30	0
2	2	3	1	40	5	40	20	45	20	45	15	50	15
2	2	3	1	50	15	50	5	40	5	Nil	Nil	Nil	Nil
3	1	3	1	0	30	0	60	30	60	30	30	0	30
4	1	3	1	30	30	30	60	60	60	60	30	30	30

Figure 2: Example of geometry table for Polygon Geometry using SQL

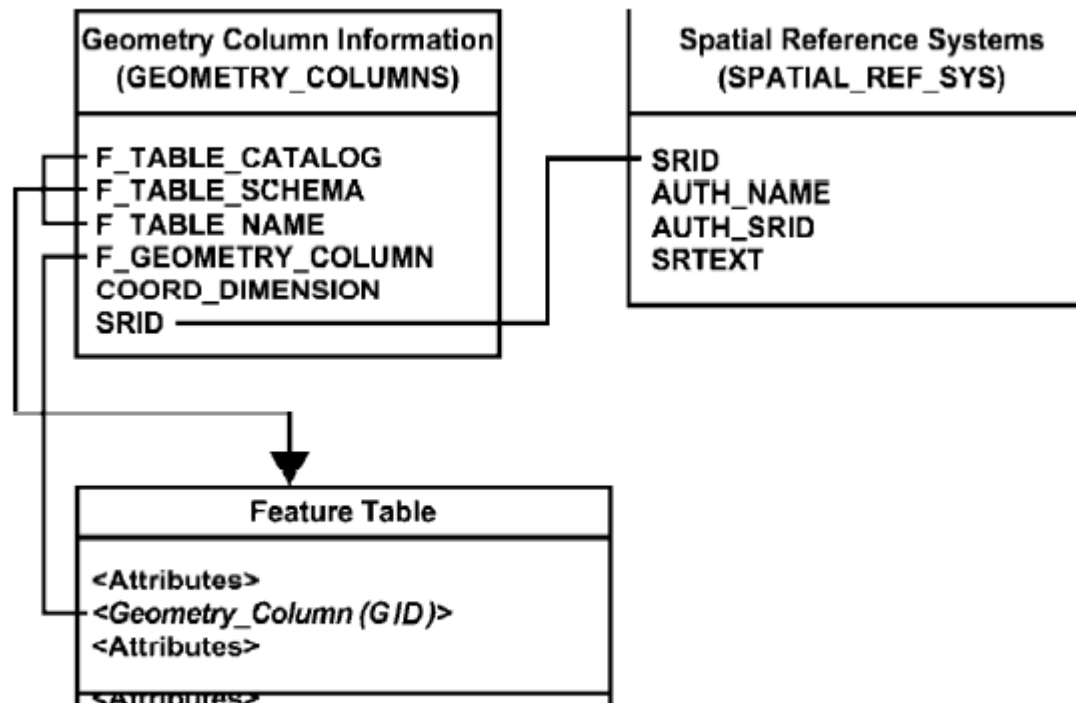


Figure 3: Schema for feature tables using SQL with Geometry Types

- ▶ UDT – User defined Data Types

- ▶ Noen (de fleste?) DBMS har støtte for dette

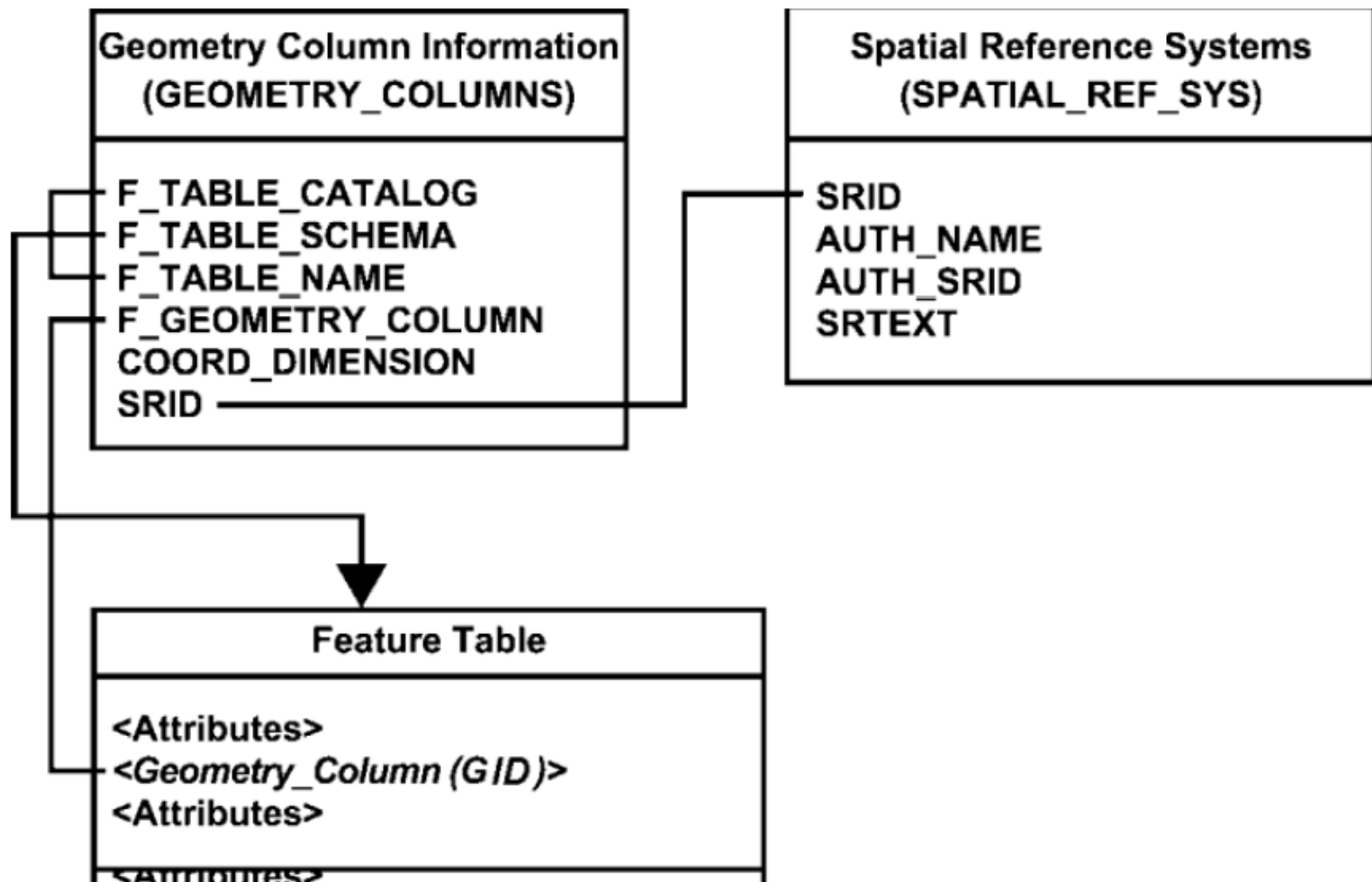


Figure 3: Schema for feature tables using SQL with Geometry Types



Egenskaper for skjematypene

▶ Normalized Geometry Schema

- ▶ Kan lagres i alle DBMS.
Kan leses av «alle», dvs. alle som kan skrive SQL
Ingen (?) GIS-program støtter dette direkte
Men: dette er et standard skjema med de fordeler det gir
Eks. på implementasjon: ArcSDE for Oracle – opp til v. xx.

▶ Binary Geometry Schema

- ▶ Kan lagres i alle DBMS.
Data må leses av et program som forstår Well Known Binary Representation for Geometry
Eks. på implementasjon: ArcSDE/ArcGIS for Server

▶ Geometry Type Schema

- ▶ Krever at DBMS har støtte for UDT, eventuelt har innebygd Geometry Type.
- ▶ Eks. på implementasjon: ArcGIS for server (Oracle, ...), PostGIS

Fra PostgreSQL til PostGIS

- ▶ Legger til datatyper
- ▶ Legger til predikater/ operasjoner/ funksjoner
- ▶ (Legger til indekser)

PostGIS

► Geometry Type definert ved hjelp av SQL

```
CREATE TYPE geometry (  
    internallength = variable,  
    input = geometry_in,  
    output = geometry_out,  
    send = geometry_send,  
    receive = geometry_recv,  
    typmod_in = geometry_typmod_in,  
    typmod_out = geometry_typmod_out,  
    delimiter = ':',  
    alignment = double,  
    analyze = geometry_analyze,  
    storage = main  
);
```

PostGIS

- ▶ Funksjoner definert ved hjelp av SQL
 - ▶ Oppretter link til eksternt programbibliotek

```
CREATE OR REPLACE FUNCTION ST_Union(geom1 geometry, geom2 geometry)
  RETURNS geometry
  AS '$libdir/postgis-2.1','geomunion'
  LANGUAGE 'c' IMMUTABLE STRICT;
```

- ▶ Noen funksjoner kommer fra GEOS
 - ▶ Geometry Engine, Open Source
 - ▶ <http://trac.osgeo.org/geos/>
- ▶ Som igjen kommer fra JTS Topology Suite
 - ▶ <http://tsusiatsoftware.net/jts/main.html>

Gode grunner for å lære om geografiske databaser

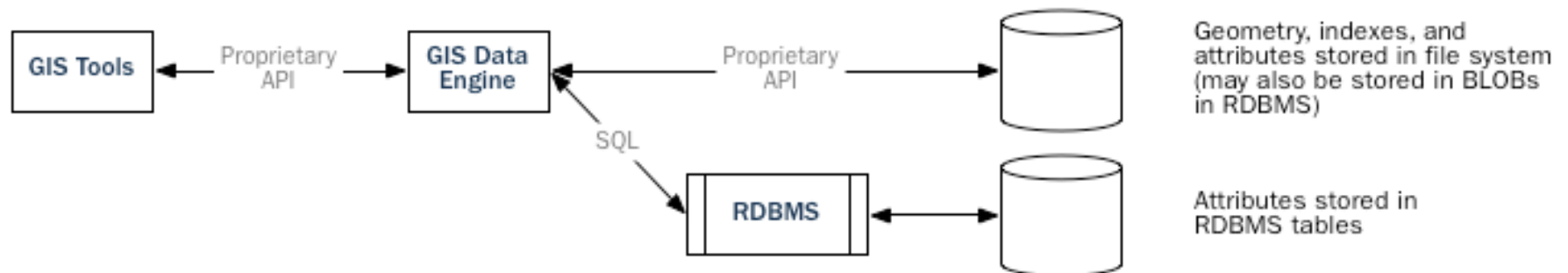
- ▶ **GeoDB er anbefalt lagringsmetode for større datasett**
 - ▶ Sentral lagring:
 - ▶ Bedre rutiner for
 - Tilgangskontroll
 - Versjonshåndtering
 - Backup
 - ▶ Standard IKT-verktøy kan brukes
- ▶ **GeoDB har stor utbredelse**
 - ▶ Viktig på svært mange arbeidsplasser
- ▶ **GeoDB og SQL fungerer bra sammen på websider**
- ▶ **SQL kan lagres i tekstfiler – (menyvalg og museklikk kan ikke) – kan dermed lagre arbeidsflyten i SQL**

Evolution of GIS Architectures

First-Generation GIS:



Second-Generation GIS:



Third-Generation GIS:



Hvorfor lære om PostGIS?

- ▶ PostGIS' kvalitet som geoDB er udiskutabel
 - ▶ Brukes over hele verden
 - ▶ Se f.eks. innlegg om PostGIS på FOSS4G-konferansene
 - ▶ Innlegg på Teknologiforum, 12. november 2014:

**Ingvild Nystuen og Lars Opsahl
(Skog og landskap):**

PostGIS som navet i
dataforvaltningen ved Skog og
landskap. Muligheter, utfordringer
og erfaringer.

- ▶ God ytelse:
 - Brukes av Kartverket som DBMS i WMS-tjenestene
- ▶ Fri programvare – åpen kildekode

PostGIS is open source software, released under the [GNU General Public License](#), that implements the [Open Geospatial Consortium's](#) [“Simple Features for SQL Specification”](#).

PostGIS also works well as a data source for GeoServer, which provides services like [WMS](#), [WFS](#), [WCS](#), and [WPS](#).

Hvorfor lære om OpenGeo Suite

- ▶ QGIS fungerer godt sammen med PostGIS og er også fri programvare som kan dekke en lang rekke behov for GIS programvare
- ▶ OpenGeo Suite inneholder også Geoserver som skal brukes i vår-emnet:
Infrastrukturer for stedfestet informasjon
(som WMS og WFS-server)

Hvordan lære å bruke PostGIS?

- ▶ Gå igjennom oppgaven i PostGIS Workshop fra FOSS4G 2011:
 - ▶ Introduction to PostGIS
 - ▶ <http://workshops.boundlessgeo.com/postgis-intro/>
 - ▶ Teoristoff
 - ▶ Praktiske oppgaver (med fasit)
 - ▶ Gå igjennom Section 1-14 + 23
 - ▶ Oppgaver knyttet til GLT-databasen fra ukeoppgave (kommer)
 - ▶ Se nærmere informasjon i Fronter