The ESRI Geodatabase Framework

Future developments at ArcGIS 10.2 and 11

MSc Marco Boeringa
2013

Author

MSc Marco Boeringa, May 2013

The Netherlands

Document version: V1.0

Introduction

This document and the three diagrams it contains supplement the main document titled:

"The ESRI Geodatabase Framework"

that I wrote and published as a PDF on the ESRI User Forums.

I made this supplement to account for the recently announced deprecation plans for ArcGIS 10.2, that contain some important notifications of ESRI regarding the future support for components of the ESRI software product line. As these announcements will impact the ESRI Geodatabase Framework, I thought a supplement that visualized these changes might be of help to others in understanding the upcoming changes. I visualized the changes by creating three separate diagrams, displaying the changes and features in 10.1, 10.2 and a presumed next generation ArcGIS 11. I have added comments to the diagrams (e.g. "Future") to explain changes.

In addition, I have included the new **Spatial Framework for Hadoop** in the diagram for ArcGIS 11, as it more or less extents the ESRI Geodatabase Framework with an entirely new option: to store, handle and process **Big Data** via **Apache Hadoop**. Developed by ESRI, this framework extents Hadoop with spatial functions based on OGC ST_Geometry. Although it is included in the diagram of ArcGIS 11 only, it is already available at ArcGIS 10.1 and downloadable as an OpenSource project from GitHub.

Please note this is currently particularly "developer oriented", although the Spatial Framework for Hadoop does include a more user friendly Geoprocessing Toolset, but defining actual Job tasks for this may need Java programming, or at least Hive HQL, knowledge. Additionally, setting up and managing a Hadoop cluster is *not* a trivial task, and requires Linux/Unix expertise. There are enterprise grade platforms for Hadoop that supposedly ease the tasks of managing a Hadoop cluster considerably. Two examples are MapR and WANdisco, a comprehensive overview of Hadoop platforms can be found on this Hadoop Wiki page: http://wiki.apache.org/hadoop/Distributions andCommercial Support

Since many of the concepts, terms, and software product names regarding Hadoop and "Big Data" may be unfamiliar to traditional GIS users, I have included a small glossary encompassing the most important concepts of Big Data and this new addition to the ESRI Geodatabase Framework. You can find this glossary at the end of the document.

Please note that the inclusion of this new Hadoop option, does not mean ESRI will replace, or intents to replace, the ArcSDE Technology based Enterprise Geodatabase Framework by Hadoop in the future. Nor is it a replacement for the ArcSDE *Application Server*, at which former location in the ArcGIS 11 diagram it is drawn simply for practical reasons. It is a *new* and *different* option, that may offer benefits or new opportunities for analysis in case of very specific "Big Data" questions in the scientific research community, or for business analytics on giant consumer datasets.

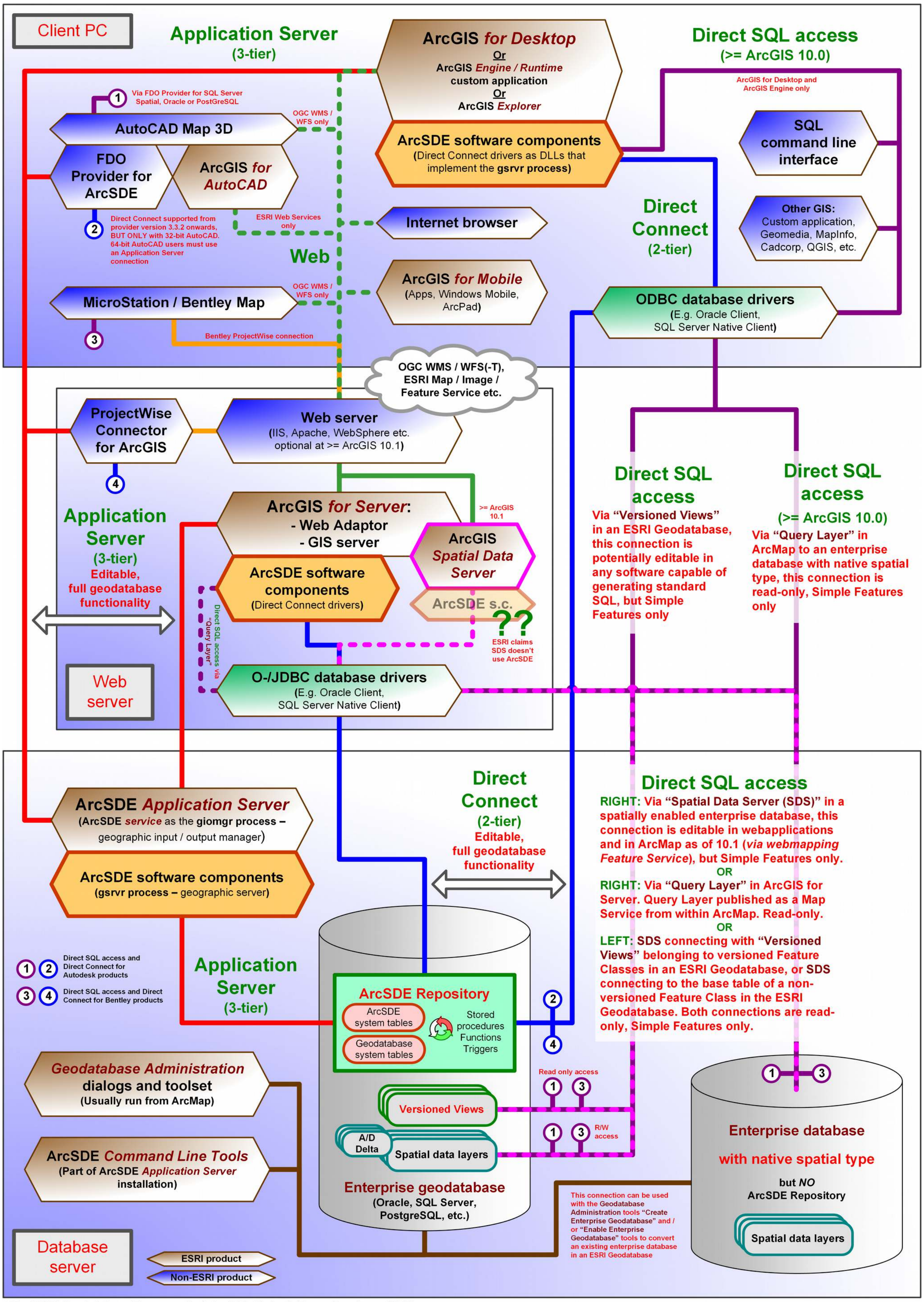
Some resources from ESRI concerning the new option:

Video: Big Data: Using ArcGIS with Apache Hadoop

Resources: http://www.esri.com/technology-topics/big-data/resources

Blog: by one of the ESRI developers involved in the Spatial Framework for Hadoop

GitHub: Spatial Framework for Hadoop



Client PC
Application Server
(3-tier)
ArcGIS for Desktop
Or
ArcGIS Engine / Runtime
custom application
Or
ArcGIS Explorer
ArcSDE software components
(Direct Connect drivers as DLLs that implement the gsrvr process)
Direct SQL access
(>= ArcGIS 10.0)
ArcGIS for Desktop and ArcGIS Engine only
SQL command line interface
Other GIS:
Custom application, Geomedia, MapInfo, Cadcorp, QGIS, etc.
1
Via FDO Provider for SQL Server Spatial, Oracle or PostGreSQL
OGC WMS / WFS only
AutoCAD Map 3D
FDO Provider for ArcSDE
ArcGIS for AutoCAD
2
ESRI Web Services only
Internet browser
Web
Direct Connect
(2-tier)
ArcGIS for Mobile
(Apps, Windows Mobile, ArcPad)
OGC WMS / WFS only
MicroStation / Bentley Map
ODBC database drivers
(E.g. Oracle Client, SQL Server Native Client)
3
Bentley ProjectWise connection
OGC WMS / WFS(-T), ESRI Map / Image / Feature Service etc.
ProjectWise Connector for ArcGIS
Web server
(IIS, Apache, WebSphere etc. optional at >= ArcGIS 10.1)
4
Application Server
(3-tier)
Editable, full geodatabase functionality
ArcGIS for Server:
- Web Adaptor
- GIS server
>= ArcGIS 10.1
ArcGIS Spatial Data Server
ArcSDE software components
(Direct Connect drivers)
ArcSDE s.c.
??
ESRI claims SDS doesn't use ArcSDE
Direct SQL access via "Query Layer"
Web server
O-/JDBC database drivers
(E.g. Oracle Client, SQL Server Native Client)
Direct SQL access
Direct SQL access
(>= ArcGIS 10.0)
ArcSDE Application Server
(ArcSDE service as the giomgr process – geographic input / output manager)
ArcSDE software components
(gsrvr process – geographic server)
Direct Connect
(2-tier)
Editable, full geodatabase functionality
Direct SQL access
1 2 Direct SQL access and Direct Connect for Autodesk products
3 4 Direct SQL access and Direct Connect for Bentley products
Application Server
(3-tier)
ArcSDE Repository
ArcSDE system tables
Geodatabase system tables
Stored procedures Functions Triggers
2
4
Geodatabase Administration dialogs and toolset
(Usually run from ArcMap)
Read only access
1 3
Versioned Views
1 3
R/W access
A/D Delta
Spatial data layers
ArcSDE Command Line Tools
(Part of ArcSDE Application Server installation)
Enterprise geodatabase
(Oracle, SQL Server, PostgreSQL, etc.)
1 3
Enterprise database with native spatial type
but NO ArcSDE Repository
Spatial data layers
Database server
ESRI product
Non-ESRI product

Direct Connect supported from provider version 3.3.2 onwards, BUT ONLY with 32-bit AutoCAD. 64-bit AutoCAD users must use an Application Server connection

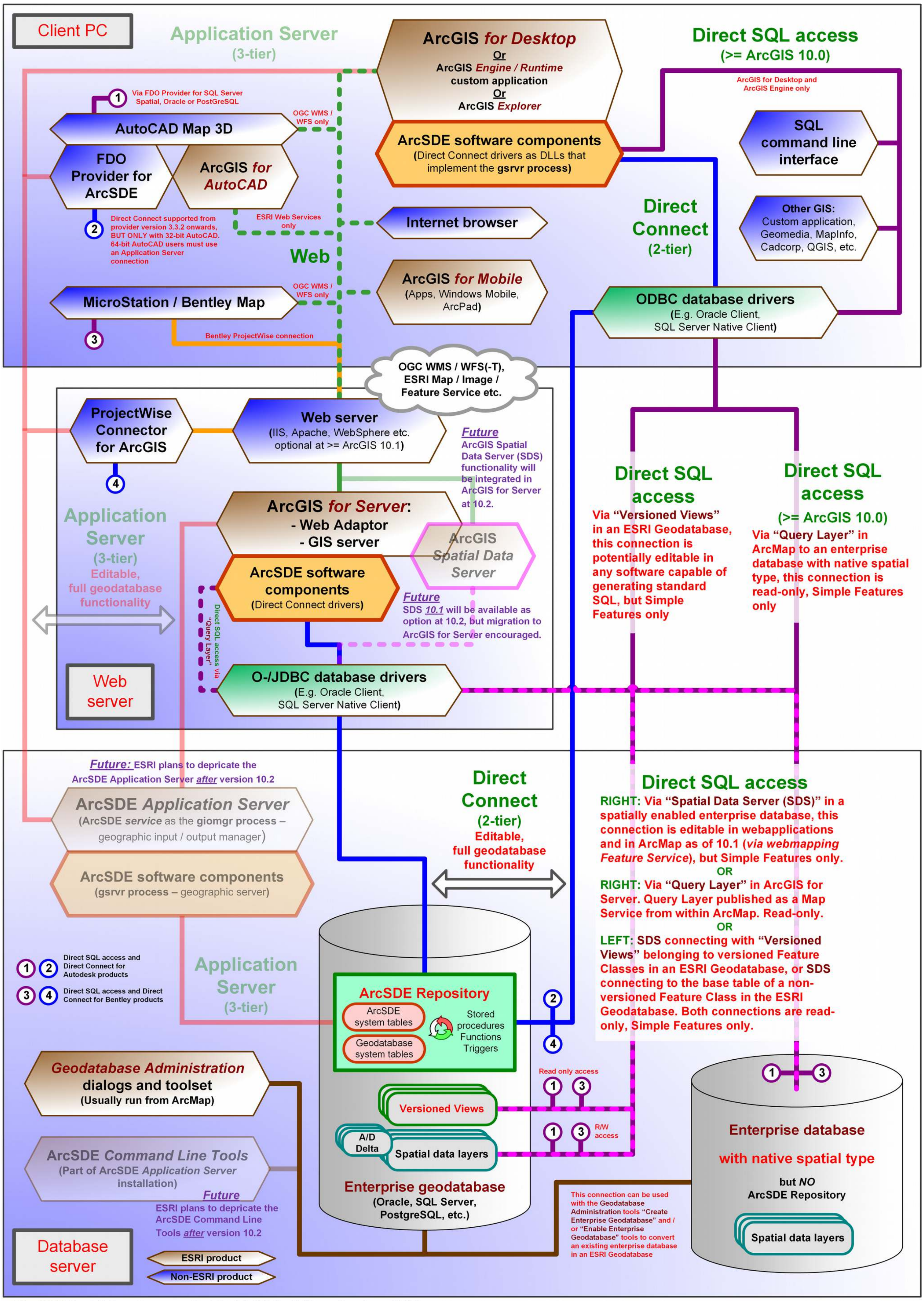
Via "Versioned Views" in an ESRI Geodatabase, this connection is potentially editable in any software capable of generating standard SQL, but Simple Features only

Via "Query Layer" in ArcMap to an enterprise database with native spatial type, this connection is read-only, Simple Features only

RIGHT: Via "Spatial Data Server (SDS)" in a spatially enabled enterprise database, this connection is editable in webapplications and in ArcMap as of 10.1 (*via webmapping Feature Service*), but Simple Features only.
OR
RIGHT: Via "Query Layer" in ArcGIS for Server. Query Layer published as a Map Service from within ArcMap. Read-only.
OR
LEFT: SDS connecting with "Versioned Views" belonging to versioned Feature Classes in an ESRI Geodatabase, or SDS connecting to the base table of a non-versioned Feature Class in the ESRI Geodatabase. Both connections are read-only, Simple Features only.

This connection can be used with the Geodatabase Administration tools "Create Enterprise Geodatabase" and / or "Enable Enterprise Geodatabase" tools to convert an existing enterprise database in an ESRI Geodatabase

The ESRI Geodatabase Framework at ArcGIS 10.1, © Marco Boeringa, 2013


Client PC
Application Server
(3-tier)
ArcGIS for Desktop
Or
ArcGIS Engine / Runtime
custom application
Or
ArcGIS Explorer
Direct SQL access
(>= ArcGIS 10.0)
ArcGIS for Desktop and ArcGIS Engine only
1
Via FDO Provider for SQL Server Spatial, Oracle or PostGreSQL
OGC WMS / WFS only
AutoCAD Map 3D
ArcSDE software components
(Direct Connect drivers as DLLs that implement the gsrvr process)
SQL command line interface
FDO Provider for ArcSDE
ArcGIS for AutoCAD
Direct Connect
(2-tier)
2
ESRI Web Services only
Internet browser
Other GIS:
Custom application, Geomedia, MapInfo, Cadcorp, QGIS, etc.
Web
OGC WMS / WFS only
ArcGIS for Mobile
(Apps, Windows Mobile, ArcPad)
MicroStation / Bentley Map
ODBC database drivers
(E.g. Oracle Client, SQL Server Native Client)
3
Bentley ProjectWise connection
OGC WMS / WFS(-T), ESRI Map / Image / Feature Service etc.
ProjectWise Connector for ArcGIS
Web server
(IIS, Apache, WebSphere etc. optional at >= ArcGIS 10.1)
4
Direct SQL access
Direct SQL access
Application Server
(3-tier)
Editable, full geodatabase functionality
ArcGIS for Server:
- Web Adaptor
- GIS server
ArcGIS Spatial Data Server
(>= ArcGIS 10.0)
ArcSDE software components
(Direct Connect drivers)
Direct SQL access via "Query Layer"
Web server
O-/JDBC database drivers
(E.g. Oracle Client, SQL Server Native Client)
Direct Connect
(2-tier)
Editable, full geodatabase functionality
Direct SQL access
ArcSDE Application Server
(ArcSDE service as the giomgr process – geographic input / output manager)
ArcSDE software components
(gsrvr process – geographic server)
1 2 Direct SQL access and Direct Connect for Autodesk products
3 4 Direct SQL access and Direct Connect for Bentley products
Application Server
(3-tier)
ArcSDE Repository
ArcSDE system tables
Geodatabase system tables
Stored procedures Functions Triggers
2
4
Read only access
1
3
Versioned Views
1
3
R/W access
A/D Delta
Spatial data layers
1
3
Enterprise database
with native spatial type
but NO ArcSDE Repository
Spatial data layers
Geodatabase Administration dialogs and toolset
(Usually run from ArcMap)
ArcSDE Command Line Tools
(Part of ArcSDE Application Server installation)
Enterprise geodatabase
(Oracle, SQL Server, PostgreSQL, etc.)
Database server
ESRI product
Non-ESRI product

Direct Connect supported from provider version 3.3.2 onwards, BUT ONLY with 32-bit AutoCAD. 64-bit AutoCAD users must use an Application Server connection

Future
ArcGIS Spatial Data Server (SDS) functionality will be integrated in ArcGIS for Server at 10.2.

Future
SDS *10.1* will be available as option at 10.2, but migration to ArcGIS for Server encouraged.

Via "Versioned Views" in an ESRI Geodatabase, this connection is potentially editable in any software capable of generating standard SQL, but Simple Features only

Via "Query Layer" in ArcMap to an enterprise database with native spatial type, this connection is read-only, Simple Features only

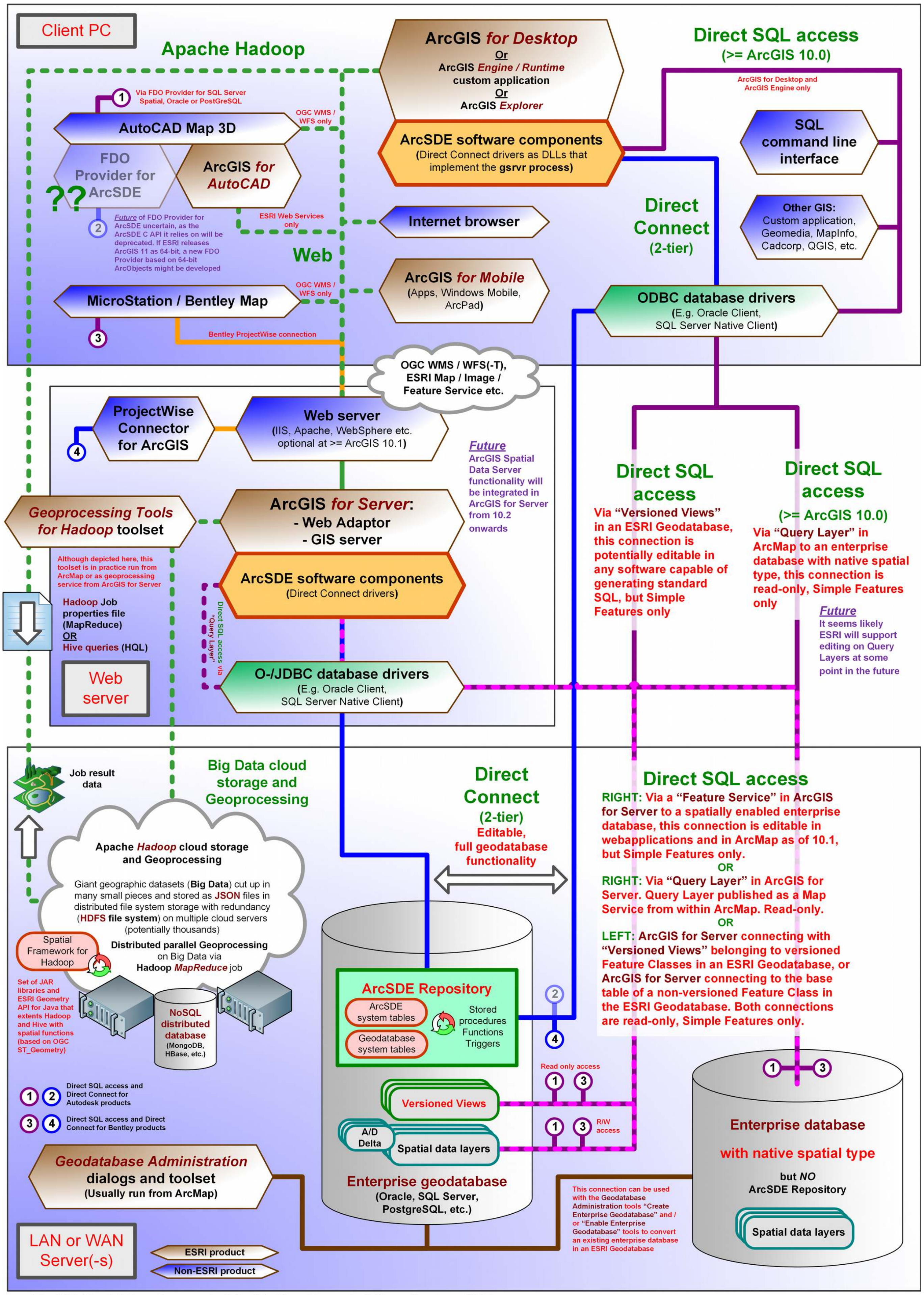
Future: ESRI plans to depricate the ArcSDE Application Server *after* version 10.2

RIGHT: Via "Spatial Data Server (SDS)" in a spatially enabled enterprise database, this connection is editable in webapplications and in ArcMap as of 10.1 (*via webmapping Feature Service*), but Simple Features only.
OR
RIGHT: Via "Query Layer" in ArcGIS for Server. Query Layer published as a Map Service from within ArcMap. Read-only.
OR
LEFT: SDS connecting with "Versioned Views" belonging to versioned Feature Classes in an ESRI Geodatabase, or SDS connecting to the base table of a non-versioned Feature Class in the ESRI Geodatabase. Both connections are read-only, Simple Features only.

This connection can be used with the Geodatabase Administration tools "Create Enterprise Geodatabase" and / or "Enable Enterprise Geodatabase" tools to convert an existing enterprise database in an ESRI Geodatabase

Future
ESRI plans to depricate the ArcSDE Command Line Tools *after* version 10.2

The ESRI Geodatabase Framework at ArcGIS 10.2, © Marco Boeringa, 2013


Client PC
Apache Hadoop
ArcGIS for Desktop
Or
ArcGIS Engine / Runtime
custom application
Or
ArcGIS Explorer
ArcSDE software components
(Direct Connect drivers as DLLs that implement the gsrvr process)
Direct SQL access
(>= ArcGIS 10.0)
ArcGIS for Desktop and ArcGIS Engine only
SQL command line interface
Other GIS:
Custom application, Geomedia, MapInfo, Cadcorp, QGIS, etc.
1
Via FDO Provider for SQL Server Spatial, Oracle or PostGreSQL
OGC WMS / WFS only
AutoCAD Map 3D
FDO Provider for ArcSDE
??
ArcGIS for AutoCAD
2
ESRI Web Services only
Internet browser
Web
Direct Connect
(2-tier)
ArcGIS for Mobile
(Apps, Windows Mobile, ArcPad)
OGC WMS / WFS only
MicroStation / Bentley Map
3
Bentley ProjectWise connection
ODBC database drivers
(E.g. Oracle Client, SQL Server Native Client)
OGC WMS / WFS(-T), ESRI Map / Image / Feature Service etc.
ProjectWise Connector for ArcGIS
4
Web server
(IIS, Apache, WebSphere etc. optional at >= ArcGIS 10.1)
Direct SQL access
Direct SQL access
(>= ArcGIS 10.0)
Geoprocessing Tools for Hadoop toolset
ArcGIS for Server:
- Web Adaptor
- GIS server
ArcSDE software components
(Direct Connect drivers)
Direct SQL access via "Query Layer"
Hadoop Job properties file (MapReduce)
OR
Hive queries (HQL)
Web server
O-/JDBC database drivers
(E.g. Oracle Client, SQL Server Native Client)
Job result data
Big Data cloud storage and Geoprocessing
Direct Connect
(2-tier)
Editable, full geodatabase functionality
Direct SQL access
Apache Hadoop cloud storage and Geoprocessing
Spatial Framework for Hadoop
NoSQL distributed database
(MongoDB, HBase, etc.)
ArcSDE Repository
ArcSDE system tables
Geodatabase system tables
Stored procedures
Functions
Triggers
2
4
Read only access
1
3
1
3
R/W access
Versioned Views
A/D Delta
Spatial data layers
Enterprise geodatabase
(Oracle, SQL Server, PostgreSQL, etc.)
1
3
Enterprise database with native spatial type
but NO ArcSDE Repository
Spatial data layers
Geodatabase Administration dialogs and toolset
(Usually run from ArcMap)
LAN or WAN Server(-s)
ESRI product
Non-ESRI product

Future of FDO Provider for ArcSDE uncertain, as the ArcSDE C API it relies on will be deprecated. If ESRI releases ArcGIS 11 as 64-bit, a new FDO Provider based on 64-bit ArcObjects might be developed

Future
ArcGIS Spatial Data Server functionality will be integrated in ArcGIS for Server from 10.2 onwards

Via "Versioned Views" in an ESRI Geodatabase, this connection is potentially editable in any software capable of generating standard SQL, but Simple Features only

Via "Query Layer" in ArcMap to an enterprise database with native spatial type, this connection is read-only, Simple Features only

Future
It seems likely ESRI will support editing on Query Layers at some point in the future

Although depicted here, this toolset is in practice run from ArcMap or as geoprocessing service from ArcGIS for Server

RIGHT: Via a "Feature Service" in ArcGIS for Server to a spatially enabled enterprise database, this connection is editable in webapplications and in ArcMap as of 10.1, but Simple Features only.
OR
RIGHT: Via "Query Layer" in ArcGIS for Server. Query Layer published as a Map Service from within ArcMap. Read-only.
OR
LEFT: ArcGIS for Server connecting with "Versioned Views" belonging to versioned Feature Classes in an ESRI Geodatabase, or ArcGIS for Server connecting to the base table of a non-versioned Feature Class in the ESRI Geodatabase. Both connections are read-only, Simple Features only.

Giant geographic datasets (**Big Data**) cut up in many small pieces and stored as **JSON** files in distributed file system storage with redundancy (**HDFS file system**) on multiple cloud servers (potentially thousands)

Distributed parallel Geoprocessing on Big Data via **Hadoop *MapReduce*** job

Set of JAR libraries and ESRI Geometry API for Java that extents Hadoop and Hive with spatial functions (based on OGC ST_Geometry)

1 2 Direct SQL access and Direct Connect for Autodesk products

3 4 Direct SQL access and Direct Connect for Bentley products

This connection can be used with the Geodatabase Administration tools "Create Enterprise Geodatabase" and / or "Enable Enterprise Geodatabase" tools to convert an existing enterprise database in an ESRI Geodatabase

The ESRI Geodatabase Framework at ArcGIS 11 and beyond, © Marco Boeringa, 2013

“Big Data” glossary

Big Data

Any (geographic) dataset big enough to choke an ordinary enterprise database server. Think of “Facebook”, “Google” type size “databases” as an extreme example. With the scaling up of the capacity of traditional enterprise databases and enhanced storage concepts for them, the point at which data can no longer be handled by an ordinary database is shifting. Still, there may be good reasons to use a distributed file storage system like Hadoop to not only store, but also take full advantage of parallel processing, of Big Data.

The Cloud

A number of server computers, also called a **cluster**, providing services over a LAN (Local Area Network) or WAN (Wide Area Network), essentially operating ***as one single unit*** to the outside world. “The Cloud” usually refers to internet based implementations of a cluster, like Amazon Elastic Compute Cloud (Amazon EC2), but it might just as well be a big server room in the cellar of your office building, or even just a dozen desktop computers hooked up in a test configuration in your section of the office. Individual servers can be ordinary commodity desktop computers, or professional rack server hardware. A service could be a distributed file storage system like Hadoop, a webmapping service, or a geoprocessing service processing data, to give some arbitrary examples.

Distributed storage and *parallel processing*

Distributed storage and *parallel (geo-)processing* using services spread (distributed) across many, potentially thousands, of “servers” on a LAN or WAN, like in “The Cloud”. This set of servers is also called a **cluster**. Since the data is processed by many different computers, distributed storage and parallel processing can potentially be much faster than traditional processing on a single server or desktop computer.

Elastic

A “buzzword” (fashionable expression) often employed when referencing The Cloud or applications running in a cloud / cluster of computers like Hadoop. “Elastic” means that individual servers / computers can be added or removed at will depending on need. This process is transparent and shouldn't affect running applications.

File System

A means to logically store and retrieve data and documents on (hard) drives. Data safety, in the sense of preventing data loss, is a key feature of many file systems, meaning some form of redundancy is build-in to cope with hardware or software failures. **HDFS** is a file system, but more commonly known are **NTFS** from Windows, or **FAT** used on digital photo cameras' memory cards. File systems can be locally, on one drive, or distributed like HDFS.

Geoprocessing Tools for Hadoop

Set of geoprocessing tools developed by ESRI that can be added to ArcToolbox to allow you to convert geographic data (e.g. a Feature Class) *to* and *from* JSON, and store the data to the HDFS file system and Hadoop. The functionality currently only supports Simple Features vector data. In addition, the Geoprocessing Tools for Hadoop allow you to start a MapReduce Job to extract data from Hadoop.

Hadoop (Apache Hadoop)

A distributed file system *and software framework* for storing and parallel processing of Big Data. Hadoop is based on **Java** and can easily run on commodity hardware. Hadoop and its underlying file system HDFS were not only designed to be able to store Big Data, but also to handle very high read/write throughput of the data from and to the distributed data store, allowing efficient processing of the data to answer "Big" questions. Hadoop is an OpenSource Apache project. Hadoop's HDFS and MapReduce are based on earlier Google work on these systems.

http://hadoop.apache.org/

http://en.wikipedia.org/wiki/Apache_Hadoop

Hive (HQL or HiveQL)

Hive, HQL (Hive Query Language) or **HiveQL**, is a query language similar to SQL for NoSQL databases. It allows querying and filtering of data to select and process data from Hadoop.

http://hive.apache.org/

Hive queries are build and submitted using command line tools, or could be defined in an ODBC client, in case one uses for an Hadoop platform like MapR, which includes an OBDC driver for Hadoop. If using Hive, there is no need for writing Java MapReduce Jobs, as extraction of data from Hadoop is guided by the HQL query statements. Hive is an extension to Apache Hadoop, and must be installed separately if not part of a pre-configured installation of Hadoop.

HDFS

HDFS (Hadoop Distributed File System) is the file system used by Hadoop. It is a distributed system, meaning data is stored on many different computers / servers, also called a **cluster**. HDFS has build-in redundancy to prevent data loss. Redundancy means that multiple copies of data (usually three copies in case of Hadoop) are stored to avoid data loss, this is also called **replication**. In combination with *automated failure detection* and *recovery*, this ensures the health of the distributed file system, as these are vital elements to make such a distributed file system viable.

HDFS can not be accessed like an ordinary drive in Windows Explorer, except through custom extensions of Hadoop. MapR seems to offer, as one of the few Hadoop implementations, a supposedly reliable NFS mounting facility for HDFS compatible with Windows, meaning you can access the HDFS file system as an ordinary drive from Windows Explorer and drag & drop data to it. A more comprehensive overview of Hadoop projects related to NFS mounting can be found here: http://wiki.apache.org/hadoop/MountableHDFS, and Google turns up more results of ongoing work to provide robust NFS mounting, e.g. Hortonworks' "Simplifying data management: NFS access to HDFS"

Job

A small program or set of instructions that will be executed against the geographic data stored in Hadoop. Jobs can be written in for example Java to extract certain data from Hadoop using the ESRI's **Spatial Framework for Hadoop** geometry functionality. Or a job could be defined using ESRI's **Hive Spatial**. There are a couple of ongoing projects attempting to provide compatibility between Python and Java, allowing Hadoop Jobs to be written in Python. Examples of this are Hadoop Streaming, Jython etc. The following pages may be of use:

A Guide to Python Frameworks for Hadoop

Jython: Python for the Java Platform

Making Hadoop Simpler with Python Joins / Keys

JSON / BSON

JavaScript Object Notation (**JSON**) is a simple text file format used to store structured and unstructured content in a hierarchical manner using key-value pairs. Just like XML, it is a human readable text format (but JSON isn't XML based).

http://www.json.org/

BSON is binary stored JSON.

Key/Value pairs (also called name/value pairs)

A "key" is essentially a unique "name" for a value to be extracted from the JSON file, or any other structured data format using key/value pairs (e.g. dictionaries in programming). By submitting the key, you get back the associated value. E.g. you could store a list of values of your friends' names using a key called "MyFriends".

An example might look like this (formatting is arbitrary here and doesn't represent JSON):

("MyFriends",{"John","Peter","Patrick","Eric"})

MapReduce

MapReduce is a programming model in the Hadoop system allowing **distributed storage** and **parallel processing** of Big Data via the Map and Reduce functions (MapReduce).

Unfortunately for GIS users, the word "Map" in **Map**Reduce has nothing to do with the traditional sense of a "map" in GIS or cartography. "Map" in MapReduce is a mechanism to cut up a big dataset stored in Hadoop in manageable smaller chunks for parallel processing and distribute and manage them in the Hadoop cluster, including performing extraction of data selections using special filtering techniques. Since Hadoop is a distributed system, instead of "tables" stored in a traditional RDBMS, selecting or filtering data is not as straight forward as going through a single table collecting records. In Hadoop and HDFS, data has to be indexed and extracted / filtered from many different servers. The Map procedure and its underlying server implementation orchestrates this process.

"Reduce" in Map**Reduce** is the possibility to parallel process or do something with the data selected by the "Map" procedure, e.g. summarize or count values. Think of calculating the, min, max or average number of households on a state or township basis.

Both the Map and Reduce function are, and must be, defined by the user in a Hadoop **Job**.

NoSQL database

A NoSQL database is a type of database that fundamentally differs from traditional relational databases (**RDBMS**). There are no "database tables" in the traditional sense, with fixed field names or content type for each field (e.g. string, date, number). A NoSQL database doesn't enforce strict relational integrity like traditional databases, but rather stores simple key-value pairs like the ones used in JSON to allow storage and extraction of data based on keys. NoSQL databases can have "SQL like" languages for extraction of values by keys. The advantages are more flexibility in the content stored, but comes at the price of less integrity assurances.

Examples of NoSQL databases are **MongoDB, HBase** etc.

Oozie

A Job tracker software or workflow scheduler allowing you to manage and submit Jobs to Hadoop in a controlled and planned manner. Oozie is Java based.

Spatial Framework for Hadoop

A "spatial extension" to Hadoop developed by ESRI based on standard **OGC ST_Geometry** allowing you to store and retrieve spatial data on Hadoop. This extension enables several spatial options in Hadoop:

- The Spatial Framework extents **Hive** with User Defined Functions (UDF), also called **Hive Spatial** by ESRI. These UDFs enable the use of standard OGC style ST_Geometry geometric functions in **Hive (HQL) queries** (e.g. ST_Polygon, ST_Within, ST_Intersection). So you can start writing HQL queries containing for example ST_Intersects, to determine if two geometries intersect.

- Addition of the ESRI Geometry API for Java. This allows you to write Java based MapReduce Job files containing spatial operators on geometries and other functions available in the ESRI Geometry API for Java (e.g. spatial indexing capability).

- A number of helper utilities or classes (**JSON Utilities**), like an **EsriFeatureClass** class object, allowing you to directly construct an ESRI Feature Class from JSON files.

The **Spatial Framework for Hadoop** can be found on this ESRI managed GitHub page:

https://github.com/Esri/spatial-framework-for-hadoop

A competitive system to the Spatial Framework for Hadoop by ESRI, is **SpatialHadoop** by the University of Minnesota. This framework is *not* an OGC ST_Geometry implementation though, like the one from ESRI, and as a consequence seems more limited in its spatial functions and types. See this page:

http://spatialhadoop.cs.umn.edu/index.html