British Columbia Geological Survey est. 1895

A geospatial frame data model to simplify digital map compilation and integration

Yao Cui

British Columbia
Ministry of Energy, Mines and Low Carbon Innovation

Paper 2021-03

Ministry of Energy, Mines and Low Carbon Innovation
Mines, Competitiveness, and Authorizations Division
British Columbia Geological Survey

Recommendation citation: Cui, Y., 2021. A geospatial frame data model to simplify digital map compilation and integration. British Columbia Ministry of Energy, Mines and Low Carbon Innovation, British Columbia Geological Survey Paper 2021-03, 20p.

A geospatial frame data model to simplify digital map compilation and integration

Yao Cui

Ministry of Energy, Mines and Low Carbon Innovation
British Columbia Geological Survey
Paper 2021-03

A geospatial frame data model to simplify digital map compilation and integration

Yao Cui[1a]

[1] British Columbia Geological Survey, Victoria, British Columbia, V8W 9N3
[a] Yao.cui@gov.bc.ca

Recommendation citation: Cui, Y., 2021. A geospatial frame data model to simplify digital map compilation and integration. British Columbia Ministry of Energy, Mines and Low Carbon Innovation, British Columbia Geological Survey Paper 2021-03, 20p.

Abstract

Digital maps in the Earth sciences have long used polygons to define features such as bedrock units. However, although useful in final products, polygons are prone to topological errors that can be time consuming to correct when data are being compiled. These errors include gaps, overlaps, and slivers at shared boundaries that are typically hard to detect and fix. Updating and integrating maps by map sheet or administrative boundaries can cause the same topological errors and introduce further problems at map borders in edge matching, such as invalid 'map sheet boundary faults' across which features falsely appear to lose continuity between adjacent sheets. Furthermore, using polygons during compilation makes it difficult to track and link generalized features in finished maps to original observations and is inefficient when updating maps at multiple scales or of different vintage. To avoid these problems, we developed a geospatial frame data (GFD) model that introduces intermediate stages between the observation and map production stages in the data lifecycle. These data compilation and integration stages are used to construct feature components from raw field or laboratory observations. Although polygons are retained in the final map product, the GFD model dispenses with polygons and topology at the compilation and integration stages. The GFD model decomposes mapped items into the most primitive and detailed feature components consisting of simple lines as feature boundaries (the 'frame', analogous to a picture frame) and centroids describing features such as geological units (analogous to a picture that fills a frame). Only a single line is used to represent linear feature components and areal boundaries, including shared boundaries. The GFD feature components are not topologically constrained. These components capture all feature types, are easy to edit, and can be stored as discrete geometric primitives in a spatial database. The GFD model allows us to develop a data checkout process to reduce map boundary problems and complex edge matching in data integration. The data checkout sets up an anchoring mechanism on the GFD feature components after extending a mapping project area to include all affected features, rather than cropping them at the limit of mapping using a cookie-cutter operation. As long as sufficient metadata at the feature-component level are available, the most detailed GFD feature components can be tracked back to original observations. The GFD feature components are thus the authoritative source data to assemble finished products. We implement the GFD model and the anchoring mechanism in PostgreSQL/PostGIS spatial database, and use *database views* and *materialized views* to operate on the GFD feature components to create polygons and assemble features in finished maps. Without the need to modify the GFD source data, the database views and materialized views can assemble data products and maps at multiple scales and level of detail or for specific purposes. Customized products can be created by reducing coordinate precision, re-projecting the map coordinate system, simplifying lines, generalizing, aggregating, and filtering feature components, and applying cartographic styling and rendering. The GFD model and integration processes can be applied to any discipline that has polygons and lines in digital map compilation and production.

Keywords: geospatial frame data, digital mapping, geology, map compilation, integration, spatial database, data quality

1. Introduction

Polygons and topology have been widely used for creating digital maps in the Earth sciences and other disciplines. Polygons were first used in Computer Aided Design software in the mid-1950s (Hurst et al., 1989a, b; Ross, 1989). In the early 1970s, a topological model of polygons was adopted by the U.S. Census Bureau and the U.S. Geological Survey (USGS) and used in the POLYVRT computer program to encode cartographic data (Peucker and Chrisman, 1975), and later in the ODYSSEY software package at the Harvard Laboratory for Computer Graphics and Spatial Analysis (Chrisman, 2006).

In geological mapping, polygons are an attractive means to capture the surface expression of three-dimensional bedrock bodies (hereafter referred to as 'areal features' for simplicity). Similarly, lines are used to portray the surface expression of quasi-planar features such as faults (hereafter referred to as 'linear features' for simplicity). However, digital maps using polygons and lines can be difficult to update and integrate. When editing or merging maps, the use of polygons can cause complex topological errors such as gaps, overlaps, slivers, and discontinuities, which are hard to detect and fix. Furthermore, if polygonal boundaries are shared with linear features, it can be laborious to reconcile the geometric differences when polygons and lines are maintained in separate map layers.

Commonly, geological surveys are conducted in predetermined areas such as map sheets specified by the National Topographic System (NTS) in Canada or the quadrangle system in the United States. A map needing updates is usually checked out of a corporate database using a 'cookie -cutter' approach in which polygons and linework are cropped at the limit of mapping. This practice introduces problems such as bedrock units being truncated with an apparent offset across adjacent map boundaries (the so called 'map sheet boundary fault' problem) when updated maps are integrated back into the corporate database, in addition to introducing topological errors in edge matching. The cookie-cutter approach can thus lead to major costs when compiling and integrating digital geology across large areas, such as at the provincial or national level.

Maps at multiple scales and data products at multiple levels of detail are required for applications such as modelling using a balanced data density, visualization at different zooming levels in geospatial web services, and cartographically enhanced hardcopy maps of different sizes, scales, and purpose. We typically treat geospatial data in a lifecycle from raw data collection to map compilation to finished products at a given scale or for a specific purpose. The map compilation stage commonly reduces observational details, making it difficult to track generalized results in finished products back to the raw data, and introducing inefficiencies when the lifecycle repeats with new observations or when a new compilation is required for a different purpose.

More than a decade ago, the British Columbia Geological Survey (BCGS) embarked on creating continuous digital coverage of bedrock geology for the entire province (Massey et al., 2005). Continuing this work, we have developed a new way to compile, update, and integrate maps that dispenses with polygons: the geospatial frame data (GFD) model. In the first part of this paper, we summarize the types of problems caused by using polygons and by mapping projects limited to individual map sheets. We then introduce the GFD model, explaining its ability to resolve many of these problems and how we have applied it to update the digital geology of British Columbia. Although we use bedrock geology as a case study, the problems and solutions are applicable to digital mapping in other disciplines, such as soil, surficial, topographic, and cadastral maps.

2. Problems in compilation and integration

In a typical geospatial data lifecycle, map compilation is the stage in which original observations are processed to produce finished maps. During this process, details are commonly generalized and, in compilations from maps created by different authors or of different vintage, significant raw observational data may be lost. Furthermore, Geographic Information System (GIS) and other drafting tools commonly use polygons to capture areal features. These practices can cause several problems in editing, updating, data integration, and production. Below we first describe the topological errors caused by rounding coordinates, shared boundaries, and edge matching in data integration. We then consider the problems with complex topological models and handling levels of detail at the map compilation stage that may result in data loss or a disconnection between finished features and their observations.

2.1. Problems caused by rounding coordinates

In digital maps, coordinates are stored using a limited number of digits. Coordinates must be truncated or rounded when their precision exceeds the minimal unit of precision for a given computer, operating system, software, or data type. Coordinate rounding can result from loading a digital map stored with high precision into a system with lower precision, re-projecting maps from one coordinate system to another, and modifying geometries (e.g., adding vertices to a geometry, moving existing vertices, splitting one geometry into multiple geometries, intersecting geometries).

Coordinate rounding can change the shape of spatial features and the topological relationships between features. For example, a decrease in unit precision (e.g., rounding coordinates from the centimetre grid to metre grid) can shift the position of lines and alter their topological relationships. As illustrated in Figure 1, line L1 changes from straight to bent at vertex V1, lines L1 and L2 change from crossing each other to being disjoined, lines L1 and L3 change from nearly touching to crossing, and lines L3 and L4 change from being disjoined to touching.

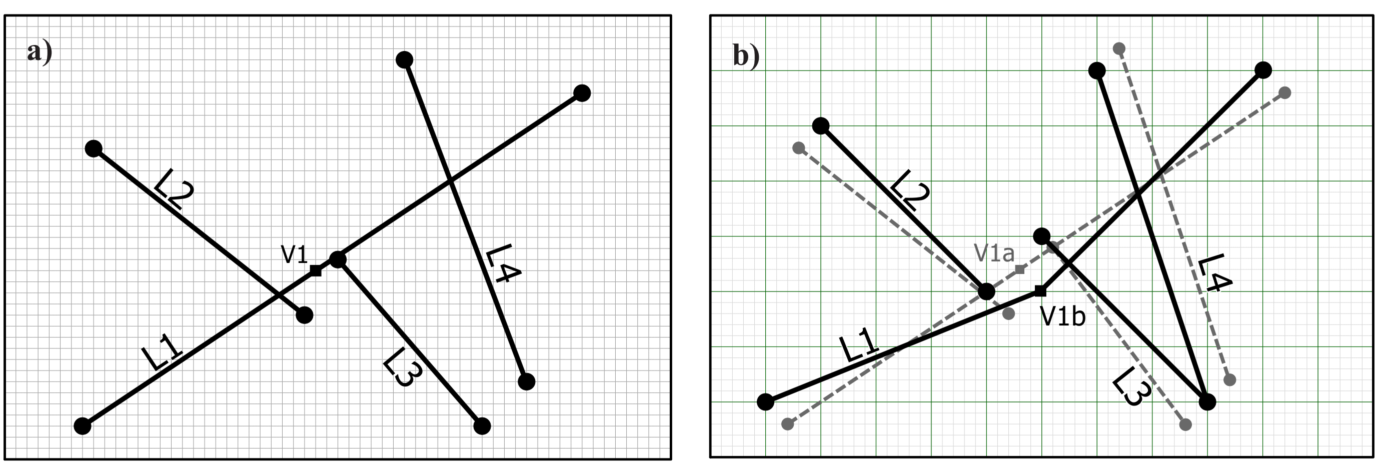

Fig. 1. Change of shape and topological relationships after rounding coordinates from centimetre to metre. **a)** Four straight lines labelled from L1 to L4, stored at a fine unit of precision (centimetre grid in grey), end of line shown as solid circles in black, and L1 with a vertex shown as square and labelled as V1. **b)** Coordinates for lines and vertex in fine precision grid (in grey) are snapped to a coarse precision grid (in green) and shown in black after rounding.

Rounding of coordinates also can occur during editing in the same precision grid, thereby changing the shape of spatial features and topological relationships between features. As illustrated in Figure 2a, L1 is a straight line that crosses L2 and is disjoined from L3. After adding a vertex V1a to L1, it snaps to the nearest grid point at V1b, which causes L1 to bend at V1b, touch L2, and cross L3 (Fig. 2b).

Splitting a line or intersecting lines can also cause coordinate rounding and thus change the shape of spatial features and topological relationships between features. As illustrated in Figure 3a, three original lines (A, B, and C) are all straight. Line A crosses B and disjoins C. After lines A and B intersect, A splits to A1 and A2, and B splits to B1 and B2. Lines A1 and A2 do not overlap with the original line A, and lines B1 and B2 do not overlap with the original line B. Intersecting A and B also caused line A1 to cross line C (Fig. 3a). Intersecting line A1 and C split line A1 to A1a and A1b, forming a node among A1a, A1b, and C (Fig. 3b). In the intersecting process, a very short line formed after splitting line C, but it became an invalid line with two identical points after snapping to the precision grid. We refer to this as a case of dimensional collapse: a geometry with higher dimension (e.g., one-dimensional line) reduced to lower dimension (e.g., points at zero dimension).

For maps with areal features captured as polygons, rounding coordinates can introduce errors not only in connectivity, but also can create overlaps, gaps, or slivers. As illustrated in Figure 4, bedrock unit B is to be subdivided into B1 and B2 (Fig. 4a and 4b). New coordinates to form boundaries between unit B1 and B2 have to snap to the precision grid (Fig. 4c), causing an overlap between unit A and [B1 and B2], and a gap between [B1 and B2] and [C and D] (Fig. 4d). If spatial intersection is applied to the map (Fig. 4d), the overlap and gap become slivers. The finer the unit of precision, the smaller the slivers and the more difficult it is to visually inspect, select, and remove them from a polygonal coverage.

Coordinate rounding can also cause dimensional collapse in polygonal coverage, resulting in changes to topology. As illustrated in Figure 5, bedrock units were originally mapped with unit C between unit B and D (Fig. 5a), and unit C touching unit A at a point. When the map is updated by adding a fault, the fault line would split units B, C, and D (Fig. 5b and 5c). The split creates new coordinates at locations N1 and N2 as part of the boundaries to form new polygons B1, B2, C1, C2, D1 and D2. However, coordinates at locations N1 and N2 are

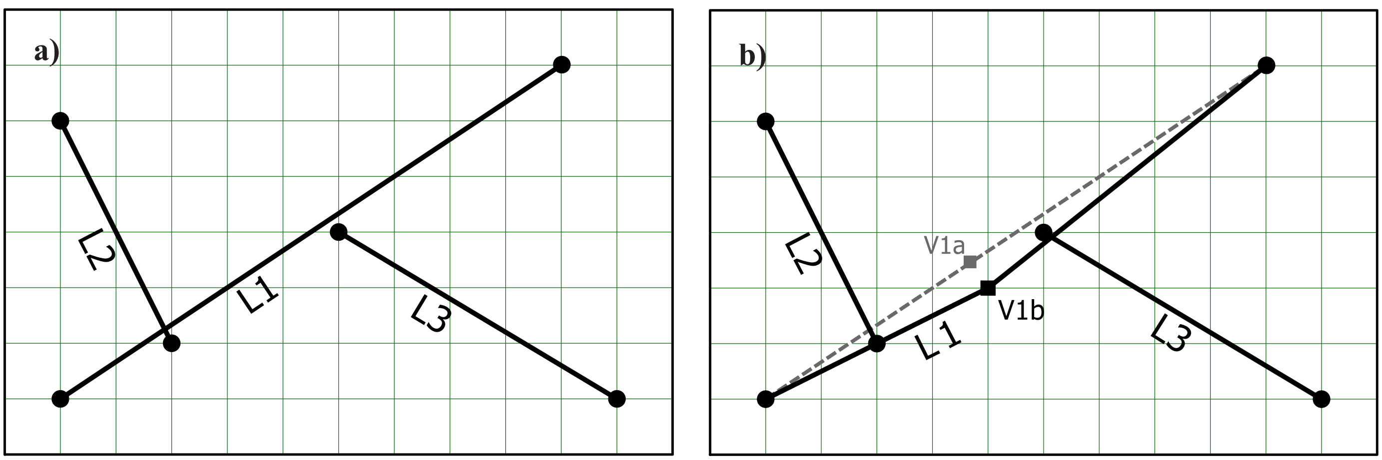

Fig. 2. Change of shape and topological relationships after adding a vertex. **a)** Three straight lines labelled from L1 to L3, stored at specific precision grid (in green), and end of line shown as solid circles in black. **b)** A vertex (V1a in grey) was added to original line L1 (dashed in grey) and the vertex is snapped to precision grid at V1b, changing the shape of line L1.

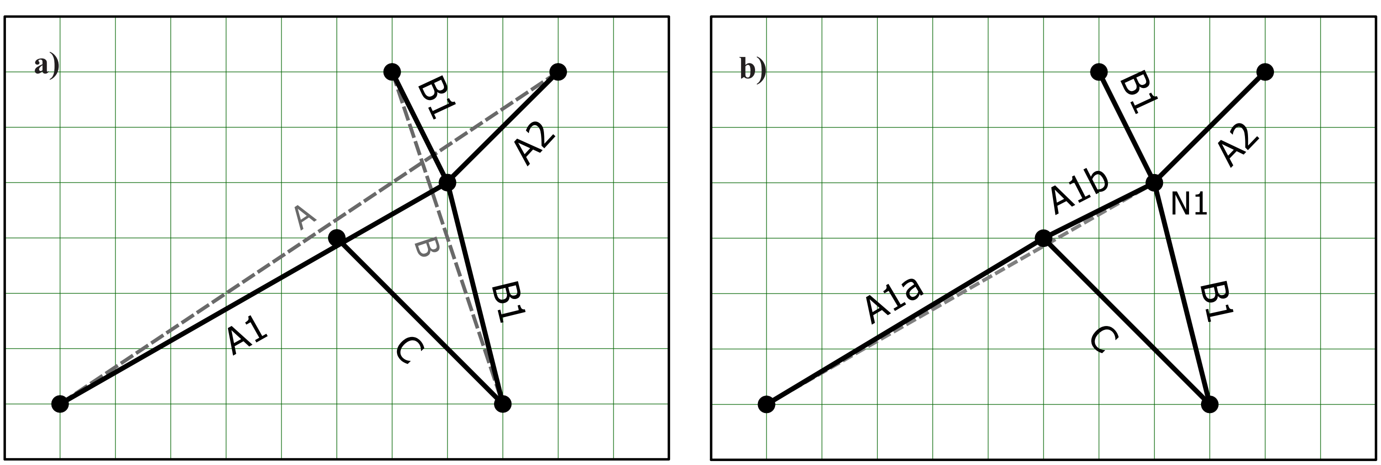

Fig. 3. Change of shape and topological relationships after intersection. **a)** Original lines A and B (dashed lines in grey) split to [A1, A2] and [B1, B2], respectively, after A intersected B. **b)** A1 (dashed line in grey) split to A1a and A1b after A1 intersected C.

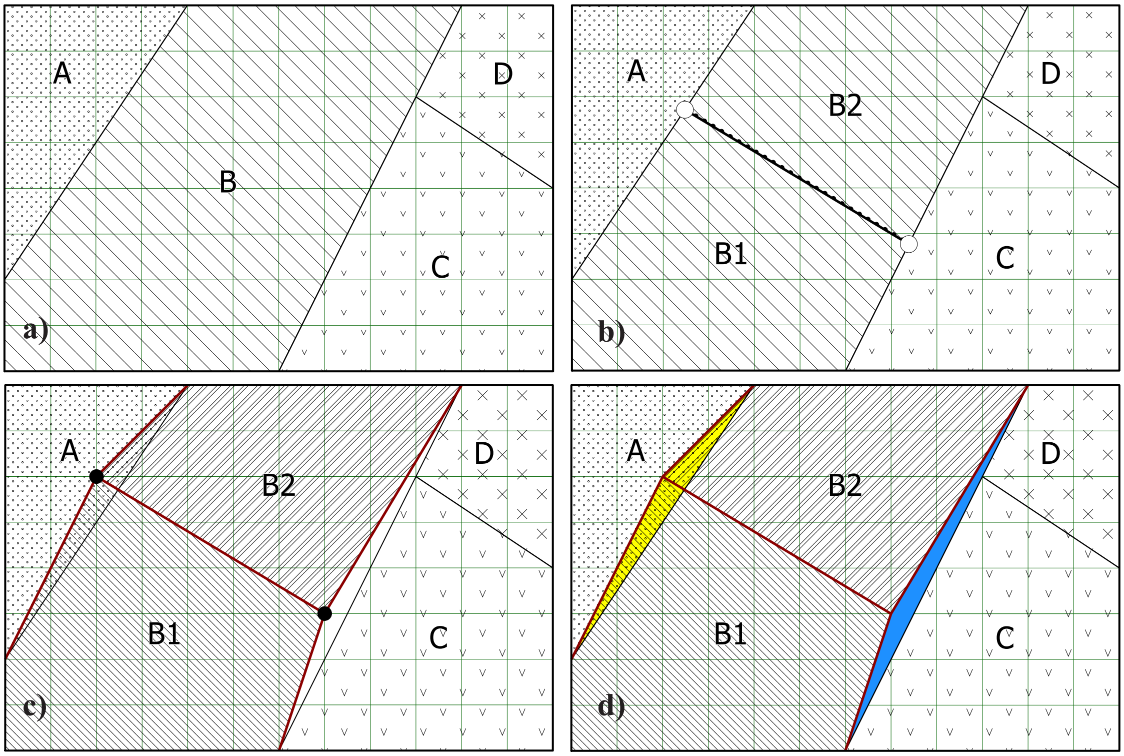

Fig. 4. Change of shapes and topological relationships among polygonal features after editing. **a)** Original map has four adjacent units as polygons stored in a specific precision grid (grid lines in grey). **b)** Map is updated to subdivide unit B to B1 and B2; new coordinates (open circles) to delineate B1 and B2 are not on precision grid. **c)** New coordinates to delineate units B1 and B2 are snapped to existing precision grid after rounding. **d)** An overlap (yellow) is formed between A and [B1 and B2], and a gap (in blue) formed between [B1, B2] and [C, D].

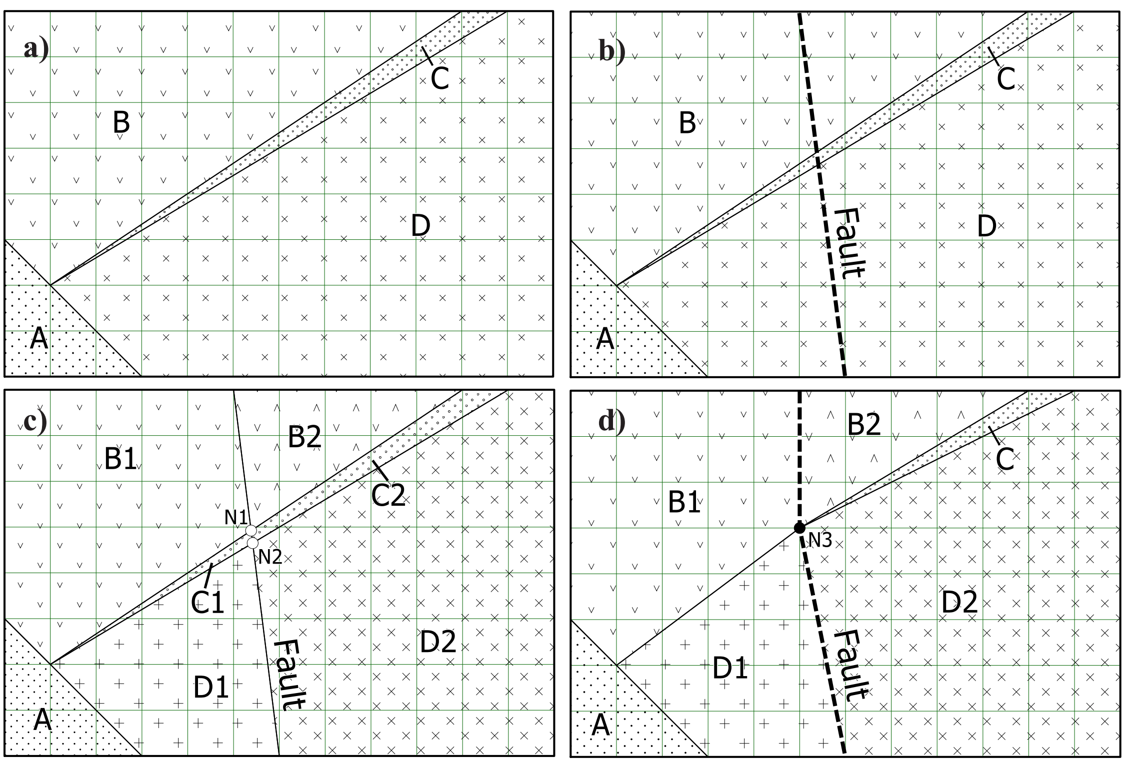

Fig. 5. Collapse of topology after splitting polygons. **a)** A bedrock map in a specified precision grid (in green) has four units (A to D) and unit C is between unit B and D. **b)** The map is to be updated by adding a fault (thick dashed line in black). **c)** New coordinates at locations N1 and N2 (open circles) are not on the precision grid in resulting polygons (B1, B2, C1, C2, D1 and D2) from split by the fault line. **d)** New coordinates of resulting polygons are snapped to the precision grid at N3 (black dot), which yields an erroneous geological relationship.

beyond the resolution of the precision grid and have to snap to the nearest grid point at location N3 (Fig. 5d). The result of rounding would cause unit C1 to become an invalid polygon (Open Geospatial Consortium, 1999) after the two coordinates for the shortest edge are snapped to the same grid location at N3. Unit C2 would become a triangle if one of the identical coordinates for the shortest edge is removed and would no longer be in direct contact with unit A (Fig. 5c and 5d). Although resulting polygons B2 and D2 remain separated by unit C, polygons B1 and D1 are now in direct contact and unit C is mistakenly portrayed as pinching out at the fault (grid location N3). Dimensional collapse happens to polygonal features along map boundaries when a map is split into multiple sheets or part of a map is cropped with a cookie-cutter operation.

In summary, digital map compilation and updating cannot avoid errors caused by coordinate rounding. In many cases, changes to the shape of linear features can be ignored if the difference is well below the resolution of a map at a given scale (e.g., line drifting less than 0.01 metre for a map at a scale of 1:20,000). Changes to the topological relationships among linear features typically introduce errors in connectivity that are easy to detect and fix, in contrast to errors in polygonal coverages. Coordinate rounding in polygonal coverage not only changes shapes, it causes dimensional collapse and thus topological errors that are harder to detect and fix.

2.2. Shared boundaries and multiple map layers

A common practice is to sort features in a two-dimensional digital map into separate data layers, first based on their geometry types, and then based on their feature types (Fig. 6). This practice has its advantages when the final map product is generated because it makes it easy to locate, display, and use specific types of features in applications. However, at the stage of map compilation, modification, and integration, this separation creates challenges in maintaining the integrity and topology between spatially related features, especially along shared boundaries of polygonal features, and between linear and polygonal features. As shown in Figure 6, bedrock unit polygons share boundaries (e.g., between units 2, 4, and 5 in data layer 4) and some of these boundaries also coincide with the boundary for mineralization (data layer 3) and a fault (data layer 2). When one of the shared boundaries is modified, it could cause geometric mismatch. In turn, the geometric mismatch results in topological errors among polygons and between polygonal and linear features, including gaps, overlaps, slivers, and discontinuities. Separating features in multiple map layers and using polygons to represent areal features impede aggregation and generalization when creating maps at multiple scales. Data integrity in finished maps may be compromised and, for maps that are regularly updated, automated generalization becomes impractical.

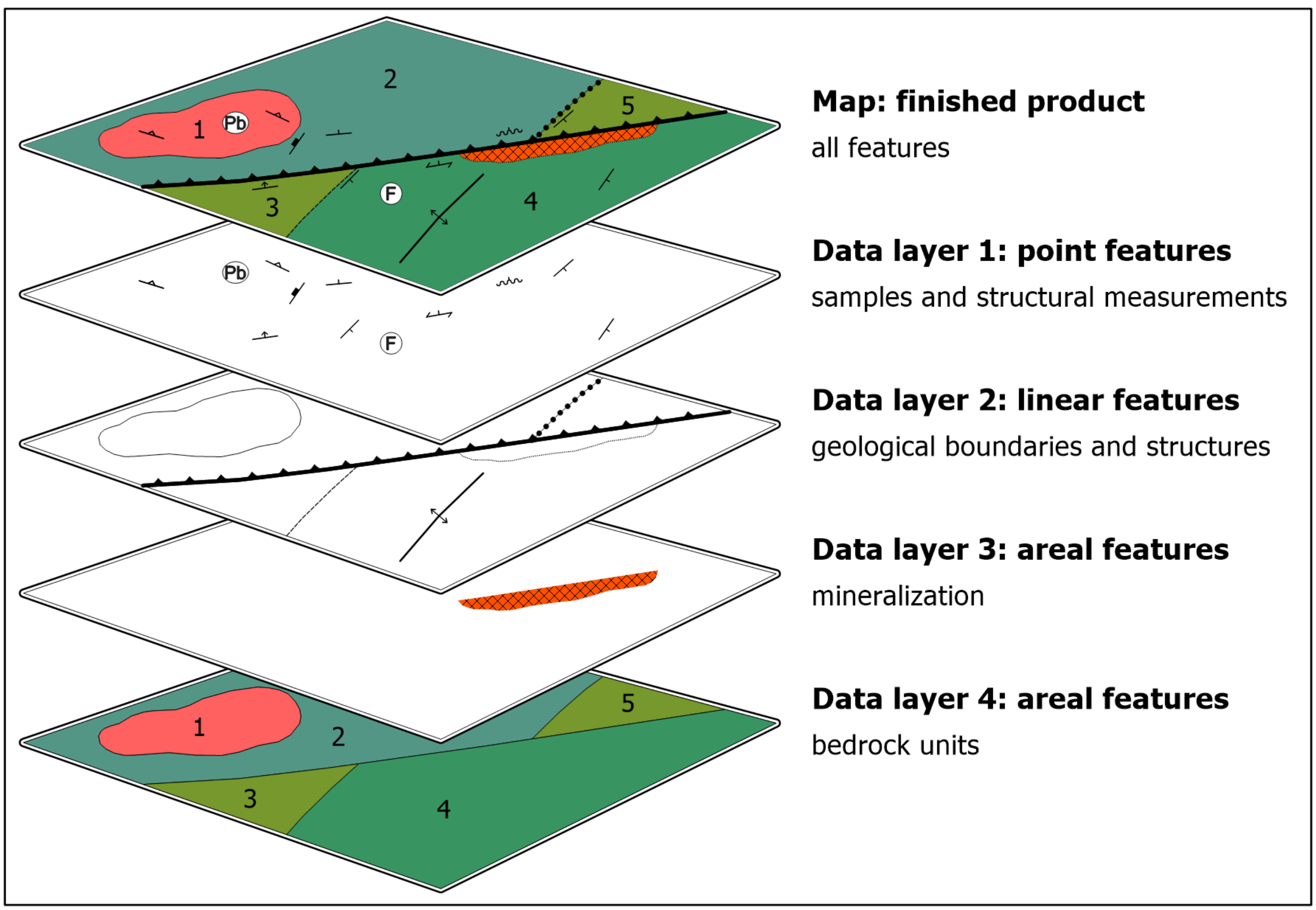

Fig. 6. An example of geological map with multiple data layers (1 to 4), organized by geometric data types and feature types.

2.3. Edge matching and map boundary problems in integration

Many national or provincial geological surveys with digital geological map coverage are faced with technical challenges when updating their corporate databases. New mapping projects commonly are restricted to predefined map sheet or administrative boundaries (Fig. 7). For example, Canada uses National Topographic System (NTS) map sheets, and the United States uses quadrangles or county boundaries. It is common practice to crop the geology for a mapping project area from a corporate database, update the geology with new work, and then integrate the updated version back into the corporate database. As described above, cropping changes geometries and topology, which introduces gaps and overlaps along map borders even if nothing has been updated. Coordinates in the cropped map area can also drift if the map has undergone transformations, such as rounding coordinates to lesser precision, loading data between GIS with different units of precision, and multiple re-projections of coordinate systems (e.g., from UTM to Albers to Lambert and back to UTM). Furthermore, new mapping, particularly if at detailed scales, can necessitate subdivision of a lithostratigraphic map unit or reassignment to a different unit. When the revised geology is integrated back into a corporate database, discrepancies at the map sheet border cause map boundary problems. In summary, map integration must deal with complex edge matching and map boundary problems if the geology is cropped from a corporate database with predetermined map sheets.

2.4. Problems with topological models and polygons

In computational geometry, underlying topological models are essential to define spatial objects and to determine the spatial relations of the objects. The widely adopted topological model is the dimensionally extended nine-intersection model (DE-9IM) developed by Egenhofer and Herring (1990), Egenhofer and Franzosa (1991), Clementini et al. (1993), and Clementini at al. (1994). There are also topological data models designed to explicitly maintain the relationships and integrity of spatial objects in GIS software, spatial databases, and certain spatial data formats, such as arc-node and edge-node-face (e.g., Peucker and Chrisman, 1975; Aronoff, 1989; Theobald, 2001). Topological rules and validation tools based on these data models have been developed to detect and fix data errors caused by using polygons and the resulting divergence of geometries at shared boundaries (e.g., Wahl, 2004; Esri, 2010). Although these measures attempt to manage topological complexity at the map compilation and integration stages, the complexity is needless and inflexible. For example, no attributes are associated with ‘edges’, and dangles are not allowed in a polygonal coverage built on the edge-node-face topological data model. Software tools can detect some errors, but in many cases manual intervention is still required to fix errors.

2.5. Compilation and levels of detail

Geospatial data typically have a lifecycle from the collection stage to the production stage, with map compilation as the process to create finished products at a given scale or for a specific purpose (and with polygons representing the surface expressions of three-dimensional features). As part of data collection in geological mapping, field surveys are carried out at different scales and, at any given scale, the actual level of detail may vary significantly. Map compilation commonly entails generalizing levels of detail and interpretation of the raw data to create a map at a given scale or for a specific purpose. This generalization reduces details to a common (generally smaller) scale, thus valuable raw data are lost. Less important features are aggregated or filtered to fit a limited space or to highlight significant items such as a map showing geological features favourable for porphyry deposits. Depending on the level of generalization, the results in the finished products can be difficult to track to the original observations. This process is inefficient when a new compilation is required to create a different product (e.g., at a different scale or for a different purpose) from the observations. For example, preliminary maps from a regional multi-year mapping project are commonly released at large scales (e.g., 1:20,000 to 1:50,000) as individual sheets of local areas but, at the end of a project, these local maps are often compiled into a single map of the entire region at a smaller scale (e.g., 1:250,000; Fig. 8). Raw data are lost in the regional map and, based on evolving information over a

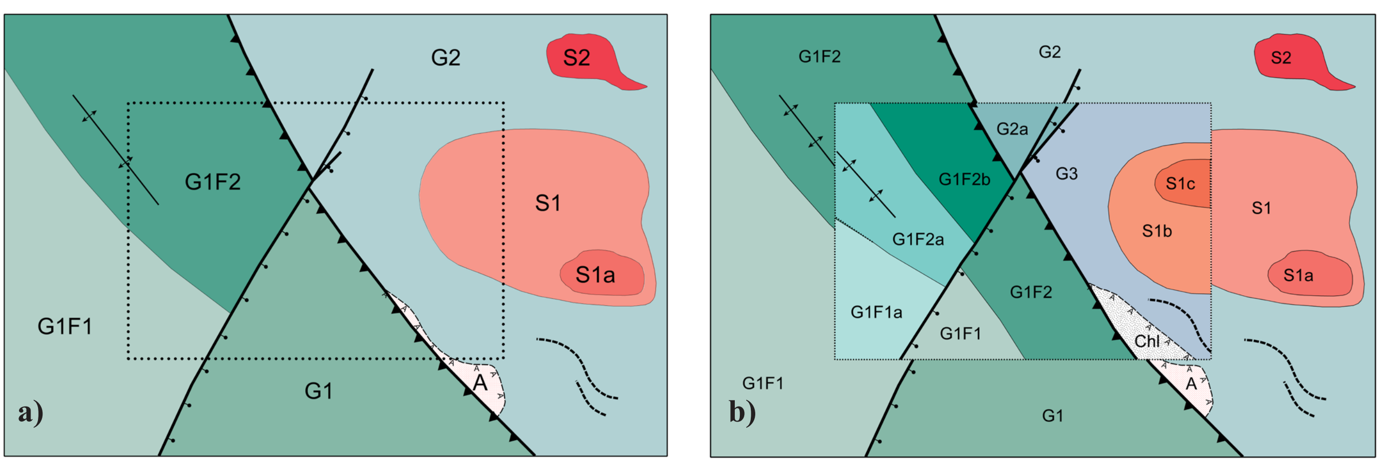

Fig. 7. a) An area defined by the limit of mapping (dotted line) is to be updated and is cut out by a cookie-cutter operation from a corporate geological map database. **b)** The updated area is merged back into the corporate database, but errors have been introduced. Map units designated by alphanumeric code (e.g., G1F2); solid line = geological contact; dashed line = dike; line with barb = thrust fault; line with ball = normal fault; line with opposing arrows = anticline; dashed line ornamented with ‘A’ = alteration.

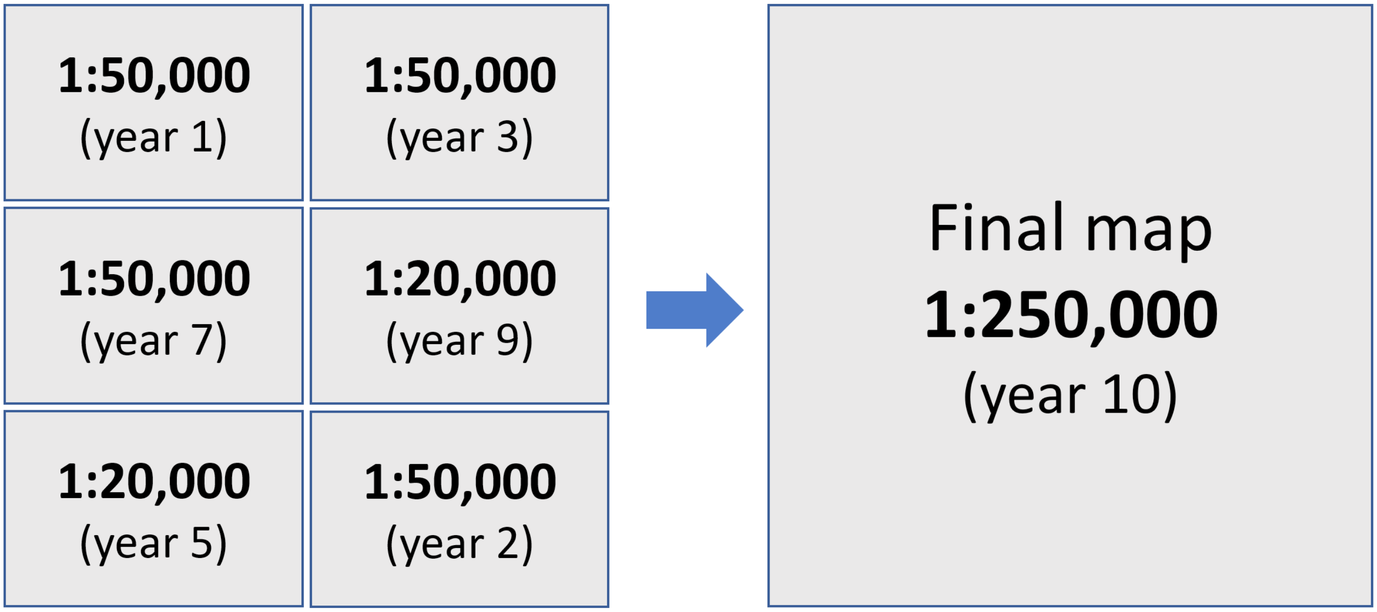

Fig. 8. The final map from a multi-year mapping project, produced at a scale 1:250,000, may lack details included in earlier preliminary maps produced at scales of 1:20,000 and 1:50,000.

period of years, the regional map and earlier maps may differ in, for example, how units are subdivided or interpreted.

In summary, we consider the root of these problems is how data are treated in the lifecycle from collection to production. New intermediate stages are required to accommodate the most primitive, detailed, and compatible feature components from the raw data. As described in the geospatial frame data model below, these intermediate stages, referred to as 'digital data compilation' and 'digital data integration', provide the ability to track and link feature components to observations (e.g., through metadata at the feature component level). Polygons representing complete features in a finished map are not created or maintained at this stage.

3. Geospatial frame data model

We developed a geospatial frame data (GFD) model to: 1) eliminate topological errors in map compilation; 2) simplify data integration without complex edge matching and map boundary problems; 3) enable tracking features back to original observations; and 4) efficiently assemble maps at multiple scales or with level of detail. In contrast to a typical geospatial data lifecycle, the GFD model adds digital data compilation and integration stages to create the most primitive and detailed GFD feature components from observations such as field surveys and laboratory analyses (Fig. 9). The GFD model compiles continuous digital coverage of the most detailed feature components available from observations, including the

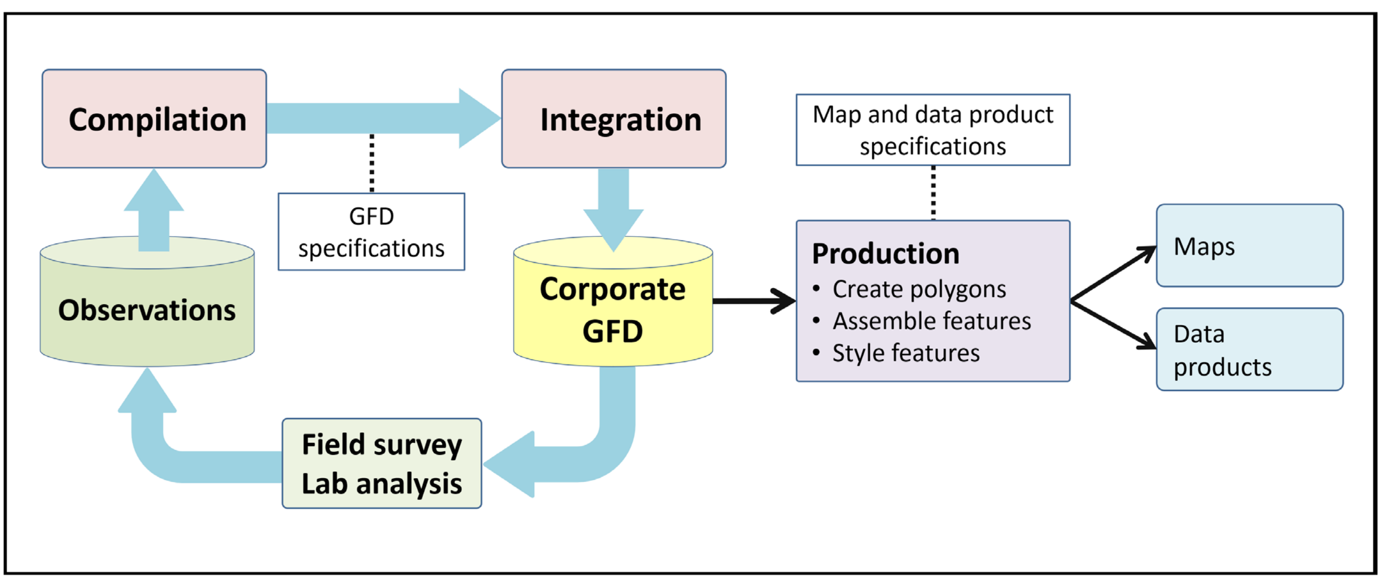

Fig. 9. The GFD model in geospatial data lifecycle with added digital data compilation and digital data integration stages. Before data integration, validation ensures that the compiled GFD feature components are compliant to the GFD specifications.

boundaries analogous to the 'frame' of a picture. A complete 'map' is the finished product built for a specific purpose or scale, and includes the 'frame' and the 'picture'.

The GFD model consists of: 1) feature components, 2) data specification, and 3) applications. Data validation ensures that the GFD feature components are compliant to the GFD specification as digitally consistent and quality product parts before data integration. Finished products, including polygons as completed features, are not part the GFD model. Applications operate on the GFD feature components as the source data (i.e., digitally consumable product parts) to customize and assemble finished maps and data products. The GFD model has no explicit topology to manage the relationship among the lines, nodes, and centroids. All linear expressions of geological features are captured and managed together as simple geometric primitives and are not topologically constrained. A line with one end dangling or both ends dangling is allowed (e.g., representing a feature in a bedrock unit in which one or both ends fail to intersect the unit boundary). As such, the GFD feature components can represent all geological feature types. The relationships among the GFD feature components can be validated during digital data compilation, and topologically constrained in the finished maps.

3.1. GFD feature components

The GFD model decomposes geological features into primitive feature components (Fig. 10). The conceptual data model for the GFD feature components is shown as a UML class diagram (Fig. 11), following the UML standards and annotation style set out by Si Alhir (1998). The feature components can be organized by: 1) simple line geometries representing parts of linear features and boundaries of areal features; and 2) point geometries or centroids containing descriptions of areal features, such as stratigraphic unit, lithology, and age. All the linear and boundary feature components are managed together in the same layer or database table. Only a single line is used to represent all linear and areal features at a shared boundary. For example, the single line shown as a thrust fault between bedrock units G1F2 and G2 in Figure 10 is also the contact that can be used as part of the boundaries to form the bedrock polygons. Feature components are modeled and organized based on a hierarchy of feature classes ('parent') and feature types ('child'). Field mappers commonly make observations at point locations; a centroid representing a bedrock unit can be linked to multiple point locations for further details or items such as samples. Multiple centroids can also be used to manage different types of areal features and level of detail (e.g., one set of centroids for mineralization and another set for stratigraphic unit).

3.2. GFD data specification and data quality assurance

In the GFD model, details for both the spatial and non-spatial feature components must be specified to ensure that they are compatible and consistent to assemble final geological maps with the required quality, integrity, behavior, and usability. In addition to common geospatial data specifications in a given discipline, the GFD uses only a single lineString that is simple and valid according to the OGC Simple Features Specification (Open Geospatial Consortium, 1999). A single line is used to represent multiple feature types using attributes or properties of the line. Complete or partial overlaps of lineStrings are not allowed. All the polygon-forming boundaries are noded at intersections. Noding at intersection is not required for linear features not sharing boundaries with areal features. Dangling features are allowed, such as faults and mineralization not intersecting other features. The absence of overlapping among lineStrings makes it possible to encode the first and second coordinates along a lineString as feature identifications that are unique and meaningful and that can be automatically generated and replicated. Explicit topological relationships are not maintained in the GFD model. The GFD feature components are meant to be modified when compiling new features, validating data quality, updating existing features, and integrating new mapping. As a result, relationships among GFD feature components are constantly broken and implicitly rebuilt based on their geological context.

Without the complexity associated with polygons, the GFD data specification requires few data quality rules to operate and validate, making it efficient. In a modern spatial database system, data types and tools are already available to store

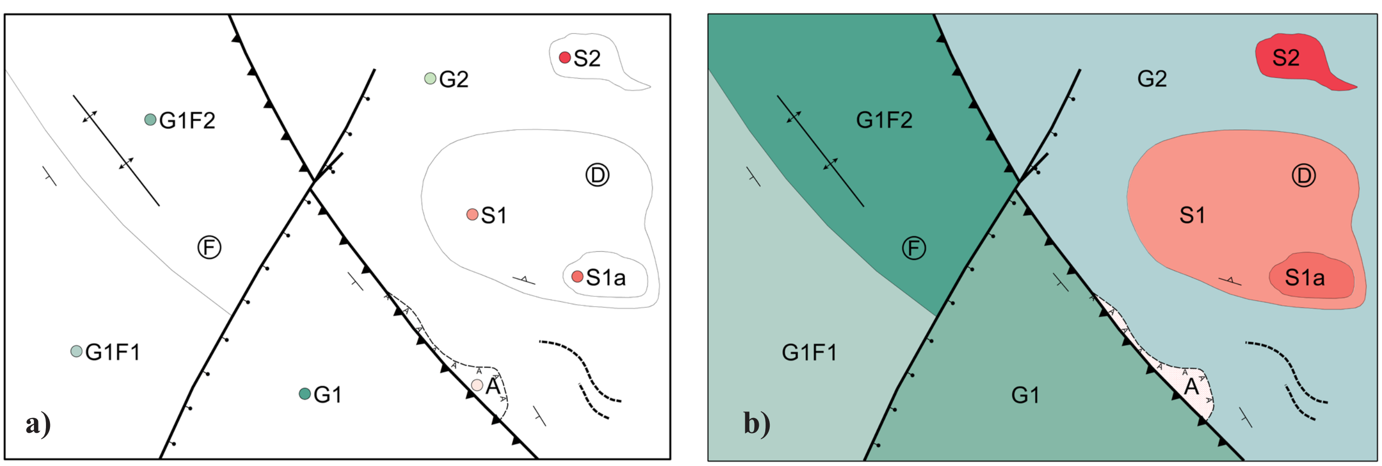

Fig. 10. a) Visual representation of feature components in geospatial frame data. Coloured circles = bedrock units; solid line = geological contact; dashed line = dike; line with barb = thrust fault; line with ball = normal fault; line with opposing arrows = anticline; dashed line ornamented with 'A' = alteration. **b)** Visual representation of geological features created from the GFD feature components.

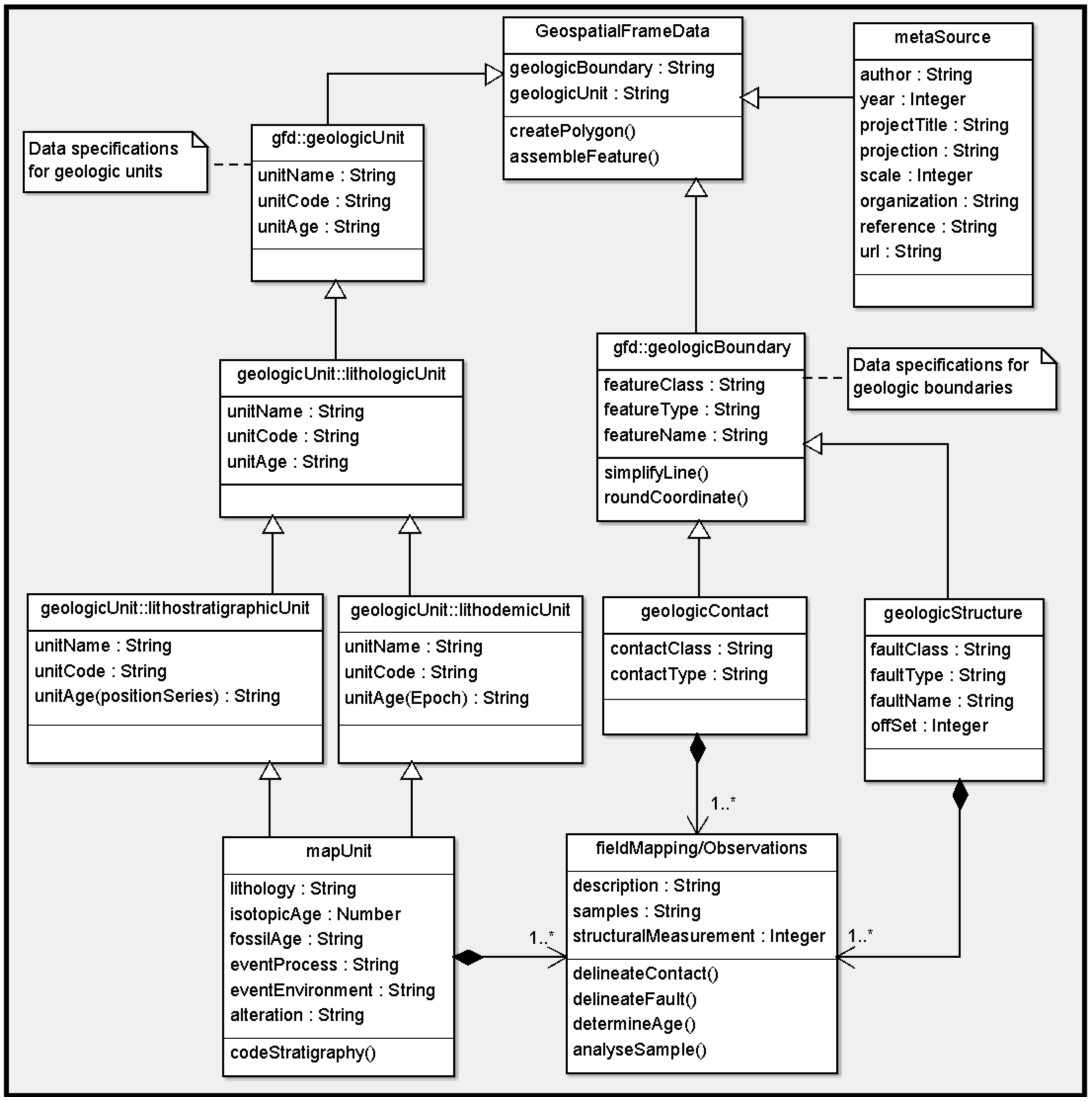

Fig. 11. Geospatial frame data model as a UML class diagram.

and operate on simple features such as geometric primitives. However, a rule-based topology system and applications can be built and operate on the GFD database to manage and validate data quality (e.g., van Oosterom et al., 2002; Martinez-Llario et al., 2017).

3.3. GFD in digital compilation

In contrast to common usage of 'compilation' by geologists when assembling a new map from raw data, the GFD model adds a new stage for digital compilation to construct GFD feature components as the detailed product parts from raw data (Fig. 12). The GFD feature components are the source data to assemble and create finished maps and data products. The GFD feature components capture the same level of detail as the original observations. The ability to track or link feature components to the observations depends on the available metadata and on operations to construct at the feature component level.

The feature components are validated against the data specification and quality rules before loading into a corporate GFD database. The GFD model and feature components make digital compilation simple by delineating geological boundaries as simple and valid lineStrings and assigning details on bedrock units to centroids. In contrast to working with polygons, common editing tasks during compilation such as adjusting, splitting, and joining lineStrings are easier to perform. Rounding coordinates of lineStrings causes fewer topological errors that are easy to detect and fix.

Data quality assurance is straightforward with simple lineString and point geometric data types. Software tools implementing OGC Simple Features Specification can detect errors in connectivity or noding by testing binary predicates such as overlaps, disjoint, and crosses between lineStrings. It is possible to validate geological relationships between the GFD feature components, even though a topology is not explicitly specified. Temporary bedrock polygons can also be constructed from the lineStrings and centroids to assist the validation. The validation processes apply error fixes to the GFD data components and the temporary bedrock polygons are deleted when validation is complete.

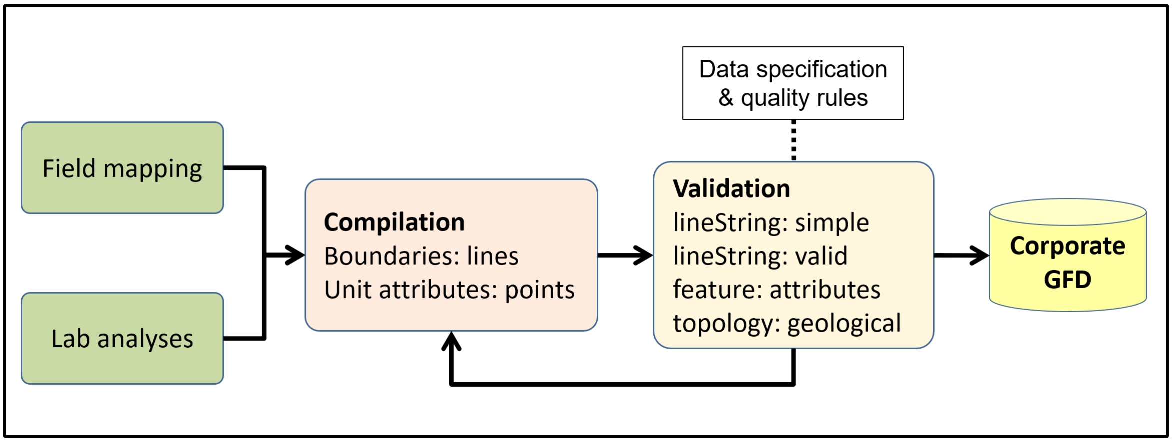

Fig 12. Workflow of GFD digital map compilation and validation.

3.4. GFD checkout and anchoring for update and integration

The GFD model allows us to develop a data 'checkout' process with an anchoring mechanism to eliminate topological errors from edge matching. This approach also significantly simplifies map integration. To update a given mapping project area (Fig. 13a) in a corporate GFD database, the GFD data 'checkout' process extracts a copy of the GFD data extended beyond the limit of mapping to include all the geological features that may be affected by the new mapping, rather than using a cookie cutter to crop features at the limit of mapping. For convenience, we recommend: 1) using the limit of mapping as a guide to intersect and select bedrock polygons created from the GFD lines and centroids; 2) dissolving the selected polygons; 3) forming a tight buffer (e.g., 1 metre) on the extended limit of mapping; and 4) using the buffer as an areal filter to select the feature components from the corporate GFD database (Fig. 13b).

Before extracting the data, we set up an anchoring mechanism to the selected feature components in the GFD database. To explain the concepts that follow, we use a few nautical terms (Fig. 14a). The GFD data intersecting the limit of mapping may be considered analogous to a boat whose position can drift and whose cargo (geological features) can be modified. We define the nodes of the outermost boundaries at the extended limit of mapping as anchor points. These anchor points maintain a fixed position. Lines that join anchor points are referred to as anchor lines. Anchor lines generally remain unmodified but can be split to add new anchor points. Lines connecting the boat to the anchor points are defined as 'rode lines'. The nodes of rode lines connecting to the anchor points can be considered as hooks latching to the anchor points. It is straightforward to develop a fully automated process to tag the GFD boundaries as 'anchor lines' and 'rode lines', and tag all the rest as 'revision' in a database (Fig. 14b). It is unnecessary to tag the conceptual anchor points and rode line hooks. Anchor points and anchor lines are shared features between those in the extended limit of mapping and beyond. The anchoring mechanism locks up anchor points in the corporate GFD database to prevent intermediate changes during new mapping; the anchor points connect rode line features from new mapping during data integration. The anchoring mechanism only works if existing anchor points are

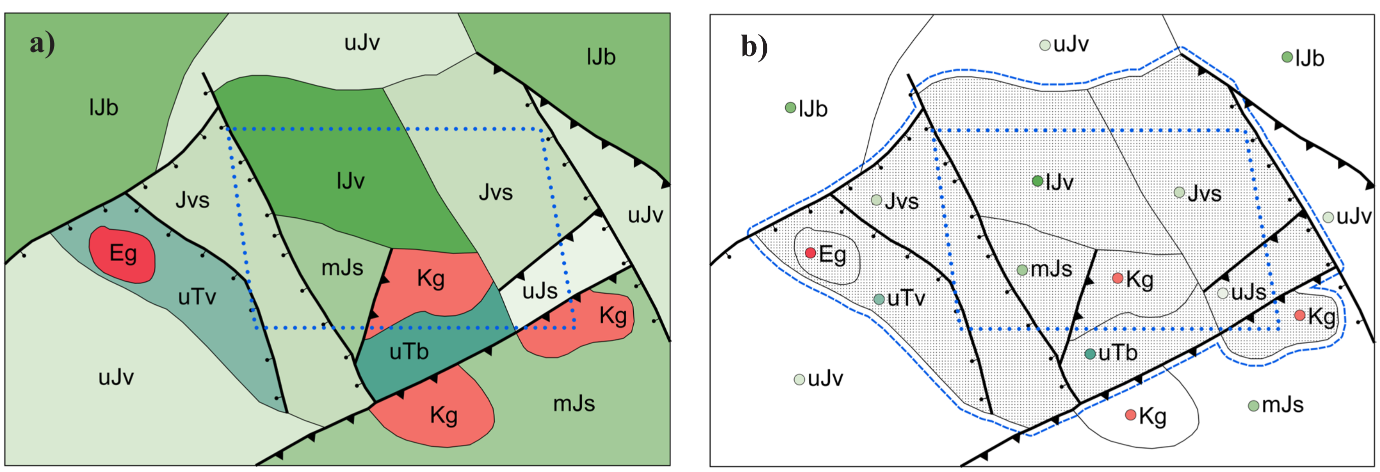

Fig. 13. GFD data checkout step 1 to extend limit of mapping and select geological features to be updated by new mapping. **a)** New mapping project area is shown as limit of mapping (dotted line in blue), overlaying geological features from corporate GFD database (lines as geological boundaries and centroids shown as bedrock unit labels), and bedrock polygons derived from the GFD features. **b)** A buffer (bound by dashed line in blue) on extended limit of mapping is used as an areal filter to select the geological features (area highlighted in grey) from a corporate GFD database.

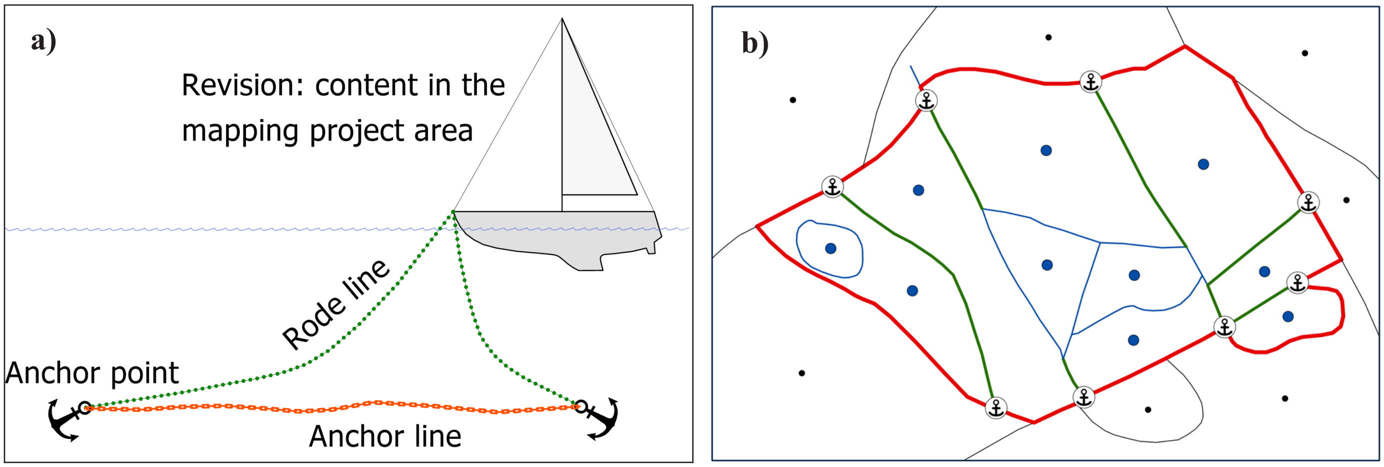

Fig. 14. Anchoring mechanism. **a)** Anchor points are guarded nodes that do not drift, anchor lines are guarded lines connecting anchor points, and rode lines connect to content for update. **b)** GFD features in the extended area are tagged by the anchoring mechanism: anchor lines in red, symbolized anchor points, rode lines in green, and all other selected GFD features tagged as 'revision' in blue.

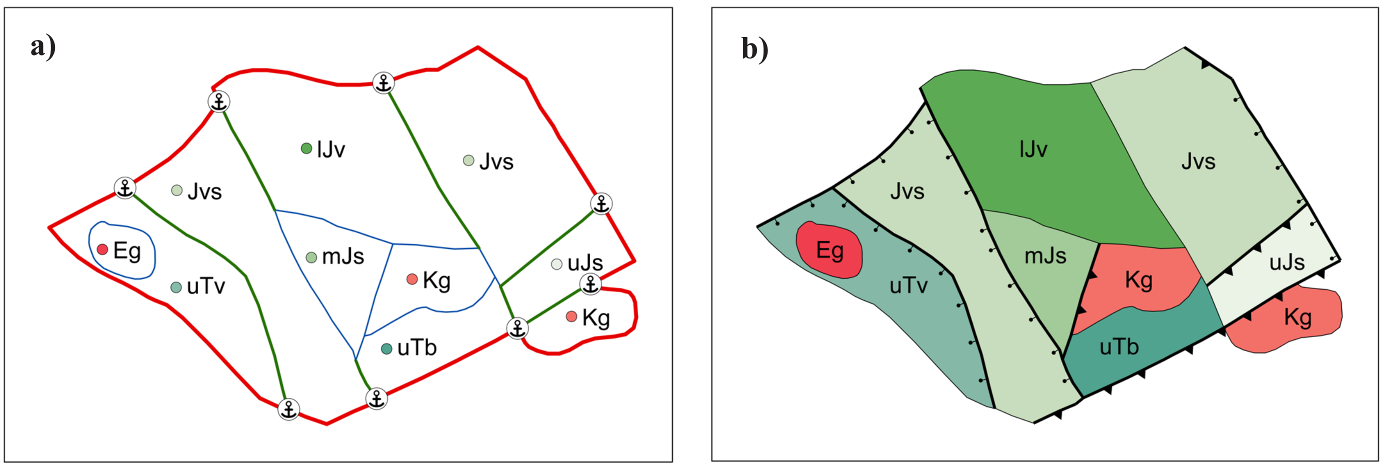

Fig. 15. a) Extracted GFD features as base for new mapping. **b)** Visual representation of styled geological features, with the addition of bedrock polygons.

not deleted or shifted beyond tolerance. Adding new anchor points to existing anchor lines is acceptable.

In the last step of GFD data checkout, a copy of the GFD feature components tagged by the anchoring mechanism is extracted from the corporate GFD database (Fig. 15a), to be used as the base for new mapping. Bedrock polygons can be created from the GFD data and provided as a visual representation of styled geological features (Fig. 15b).

When the new mapping is completed (Fig. 16a), data validation checks if: 1) the feature components are compliant to the GFD data quality specification; 2) feature components can correctly form complete bedrock polygons; and 3) any changes to anchor lines, anchor points, and new rode lines are tagged. If the anchor points and anchor lines had minimal drift and no new rode lines were added, three steps integrate the new mapping to the corporate GFD database. In the first step, GFD feature components from new mapping (except anchor lines) are loaded into the corporate GFD database (Fig. 16b). In the second step, outdated feature components in the extended area (not including the tagged anchor lines) in the corporate GFD database are tagged as 'retired' and removed to the archival database (Fig. 17a). In the last step, rode line hooks from the new mapping are snapped to the anchor points in the corporate GFD database (Fig. 17b). A simple spatial function can automatically search for the nearest anchor point within a tolerance and snap the rode line hook to the anchor point. New rode lines split anchor lines in the corporate GFD database to form new anchor points, and by default, the split process will ensure the new rode line hooks snapped to the new anchor points (Fig. 17b). No additional edge matching work is required. It is recommended to set an appropriate snap tolerance and record the snapping results (e.g., distance of adjusting rode line hooks after snapping).

The GFD data checkout process leaves no room to introduce topological errors because geological features are not cropped when data are extracted from the corporate database for updating. The anchoring mechanism replaces edge matching by simply snapping rode lines to anchor points. This process also reduces the risk of introducing discontinuities or map boundary problems in geological features, typically associated with mapping using arbitrary map borders.

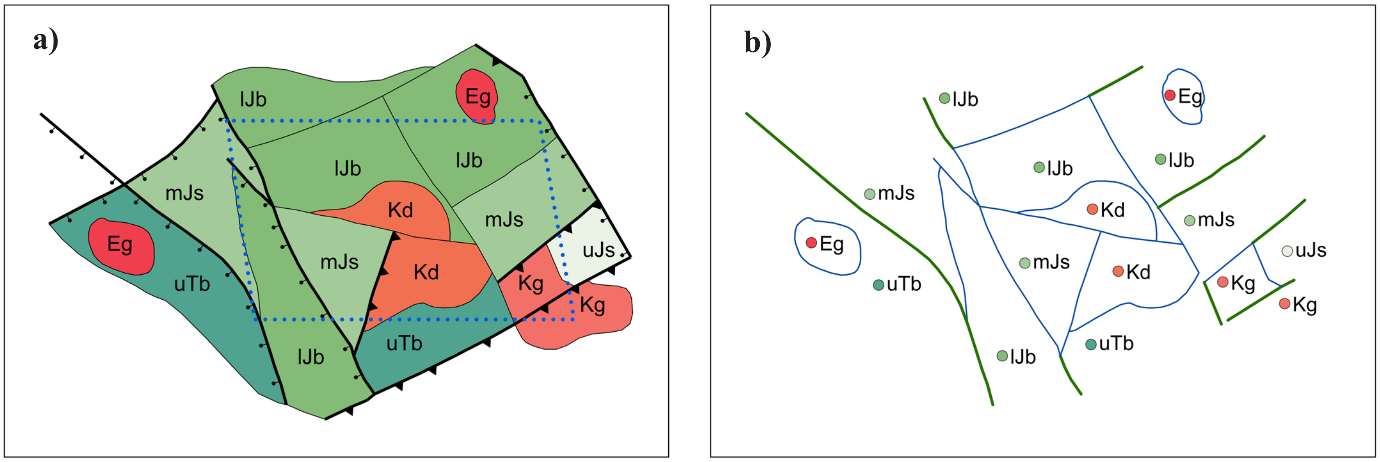

Fig. 16. GFD data integration step 1. **a)** Updated map is shown with GFD feature components and finished bedrock polygons. **b)** Required GFD feature components for integration, with new rode lines tagged and anchor lines removed.

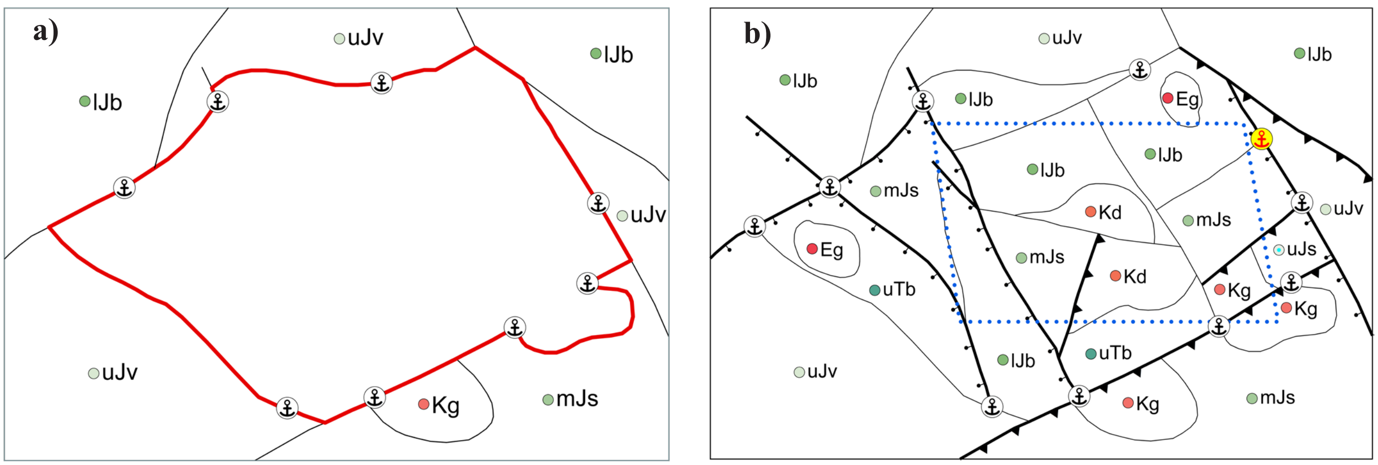

Fig. 17. Data integration step 2. **a)** Outdated feature components in the corporate GFD database are removed to archival database. **b)** Updated data are inserted into the corporate GFD database, with rode lines snapped to anchor points (in black), and new rode lines split anchor lines and forming a new anchor point (in red).

3.5. GFD to support level of details

The GFD model supports multiple levels of details not limited by specific mapping scales and can generalize source data to create finished maps at different scales without impacting the data integrity. Compared to a finished map that has a specific scale and size to present geological features at an appropriate level of detail, the GFD model is capable of accommodating feature components captured at multiple mapping scales and positional accuracies. For example, 1:50,000 fieldwork may include areas mapped at more detailed scales. We can assign every GFD feature component with two sets of scales: 1) actual mapping scale for source data, and 2) intended presentational scales or scale range for finished maps (Fig. 18a). The presentation scales determine if the GFD feature components are used or filtered out in the map-making process, including the process of forming the bedrock polygons (Fig. 18b). Rule-driven applications can assign presentational scales to feature components based on map product specifications.

Aggregation of GFD feature components is necessary to combine and extract an appropriate level of detail for the finished map at a given scale, by following the hierarchy of feature classifications, e.g., stratigraphic rank, age, lithology, and structure. Typically, we aggregate lower stratigraphic ranks to higher ranks for maps at smaller scales, and detailed rock types to more generalized ones. A rule-driven aggregation can be accomplished based on map product specifications, either as part of the map-making process, or to expand the GFD model to include complete hierarchies of stratigraphic rank, age, lithology, and structure. Aggregation and rendering rules can also be defined to produce thematic maps for special purposes, such as displaying geology favourable for porphyry deposits by ranking the importance of the geological features and assigning intensity of rendering.

Simplified versions of the geology at smaller scales can support fast display at multiple zooming levels on geospatial web services and simplified cartographic presentation on maps. We can reduce coordinate precision by rounding to the nearest metre, and then simplify the lineStrings. The map product specifications determine the level of precision reduction and lineString simplification. These processes have no impact on data integrity because they operate on the aggregated GFD feature components tagged at a presentational scale before forming bedrock polygons.

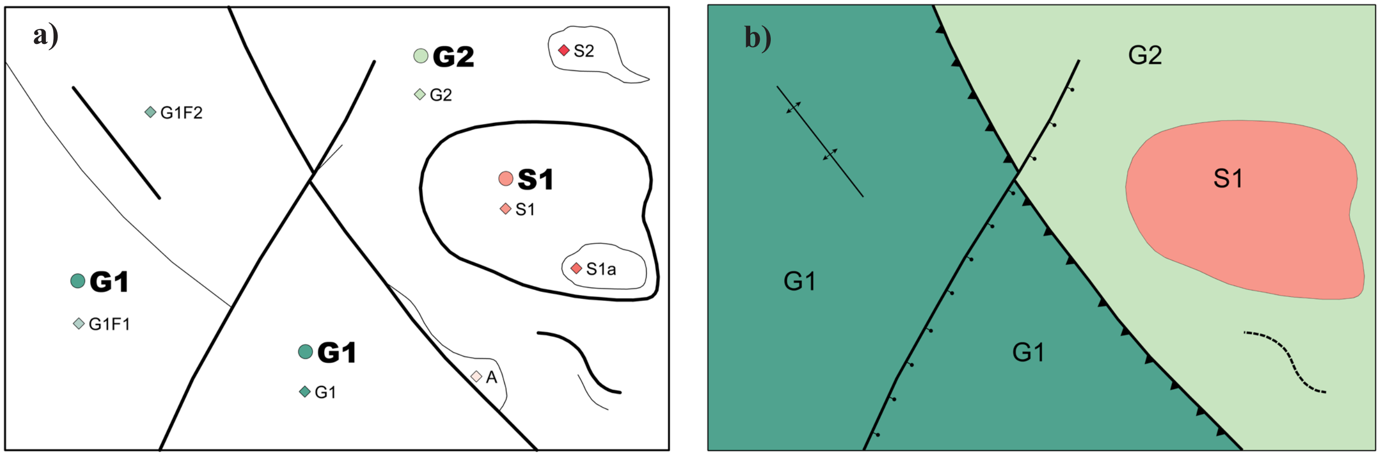

Fig. 18. Multiple levels of details and generalization of representation. **a)** All geological features are captured at a mapping scale of 1:50,000, with centroids symbolized in diamond and thin lines tagged with a presentational scale of 1:50,000, and centroids symbolized in circles tagged with a presentational scale of 1:50,000, and thick lines tagged with presentational scales of 1:50,000 and 1:500,000. **b)** A geological map at a scale of 1:500,000 is generalized from features tagged with a presentation scale of 1:500,000.

4. Implementation of geospatial frame data model

The GFD model, data checkout and anchoring mechanism can be implemented in any information system that can handle spatial data. We recommend spatial database management systems to simplify and streamline digital compilation, integration, and production for corporate geological databases. Individual geologists can benefit from the GFD concepts and principles for regional compilation in a desktop GIS environment.

4.1. GFD specifications

The GFD model requires a set of data specifications in the geometries, especially for linework (Table 1), in addition to typical data standards. We validate the GFD feature components against these specifications as data quality rules. Regardless of the implementation environment, the data validation can be carried out manually or with the assistance of software tools or functions available in desktop GIS and spatial database systems.

4.2. Desktop GIS environment

In a desktop GIS environment, the most practical use of GFD model is to compile a map only using lines representing geological boundaries and centroids representing the attributes of bedrock units. Updating, editing, and validating are always applied to the boundaries and centroids. The data specifications in Table 1 can be used to guide the compilation, manipulation, and data quality checking for the geometries.

A typical desktop GIS has geometric tools to form polygons from lines and transfer attributes from the centroids to the polygons through spatial overlay. During map compilation, polygons can be formed to test if lines are properly noded at intersections, including the matching of centroids to polygons at the last stage of data validation.

The data checkout and anchoring mechanism can be done manually or through Application Programming Interface (API) available in most open source or commercial GIS software by following the steps described in section 3.4. It is essential to add columns for the geological boundaries to accommodate data checkout tags, such as anchor line and rode line.

4.3. Corporate database environment: British Columbia Geological Survey example

A database management system offers advantages over typical desktop GIS to manage a large corporate geological database by: 1) improving performance by indexing, partitioning, and parallel processing; 2) enhancing security (authentication/permission, transactional, triggers, and back-up); 3) permitting multiple users and concurrent editing; 4) connecting by multiple clients (e.g., ODBC, OLE DB, JDBC); and 5) allowing standards-based SQL queries.

At the BCGS, we deployed the GFD model as an operational spatial database in PostgreSQL/PostGIS to update the BC digital geology (e.g., Cui et al., 2018a). We chose PostgreSQL/PostGIS because it is open-source software and one of the first to implement the OGC simple features specification and SQL Management of External Data (SQL/MED, a part of SQL standard). The GFD database extensively uses *database views* (virtual or in memory result sets of stored queries) and *materialized views* (database objects containing result sets of stored queries) in applications, in addition to database trigger functions, and stored procedures. Below we describe our implementation with script snippets as examples to illustrate some of the applications.

The entire system consists of an extended GFD model and applications for data checkout, data validation, and data check-in (integration), and production. The BCGS corporate GFD database is central to the geospatial data lifecycle (Fig. 19) and includes: 1) data checkout and anchoring; 2) new field and laboratory data collection; 3) observation database archiving; 4) digital compilation and construction of GFD feature components; 5) validation; and 6) check-in and integration back into the corporate GFD database. The GFD database contains the authoritative and most detailed feature components, possible tracing back to their observations. At the production stage, applications operate on the corporate GFD database to create various finished products as geological maps, data download, and interoperable geospatial web services.

The conceptual GFD model (Fig. 11) is extended to accommodate feature-level metadata (e.g., observation methods, history of revisions, data quality checking, and review),

Table 1. Geospatial frame data specifications on geometries.

Specification	Description
Allowable geometry	Single, simple, and valid LineString and LinearRing compliant to OGC Simple Features Specification (Open Geospatial Consortium, 1999) are allowed for planar geological features and topographic boundaries. A single point is allowed for a geological unit centroid and other point locations.
Unit of precision	Unit of precision appropriate for a given level of detail, e.g., at decimal metre or rounded to the nearest metre for coordinates in UTM, or rounded to the 6th floating point for coordinates in geographic, for a regional compilation at a scale of 1:50,000.
Duplicate and overlap	Duplication and partial overlapping of LineStrings is not allowed; LineStrings within 2 metres of each other are checked to be valid.
Noding	All polygon-forming LineStrings must be noded at intersections.
Connectivity	There must be no dangling nodes for polygon-forming LineStrings, or dangling ends of fault LineStrings near other features within a pre-determined tolerance appropriate to the level of detail (e.g., 25 metres for a regional compilation at a scale of 1:50,000).
Continuity	LineStrings are not fragmented by pseudo nodes (i.e., intersection of less than two lineStrings and no difference in attributes between the features); change of line directions (important for certain feature types such as fault types and unconformity that follow the right-hand rule in line direction) change of feature types (e.g., from contact to fault) are validated.
Minimum length of LineString	No LineStrings should be shorter than a minimum length or tolerance appropriate for a given level of detail, unless they are required to connect features.
Distance between vertices	Distance for a line segment between two sequential coordinates should be appropriate for a given level of detail (e.g., no less than 2 metres and no more than 500 metres for a regional compilation at a scale of 1:50,000).
Overshoots	Overshoots of LineStrings (e.g., fault) are removed for those shorter than a minimum length appropriate for a given level of detail.
Coordinate density	Density of coordinates should be appropriate for a given level of detail: 1) to avoid redundancy in coordinates, and 2) prevent distortion of shape from transformation with too few coordinates.
Geometric irregularity	Sharp spikes along a linear LineString are checked and removed; sharp wedges in a LinearRing are checked to be valid; sharp angles between two LineStrings are checked, validated as real, or corrected.
Data matching	Each centroid should match one resulting polygon forming from the geological boundaries.

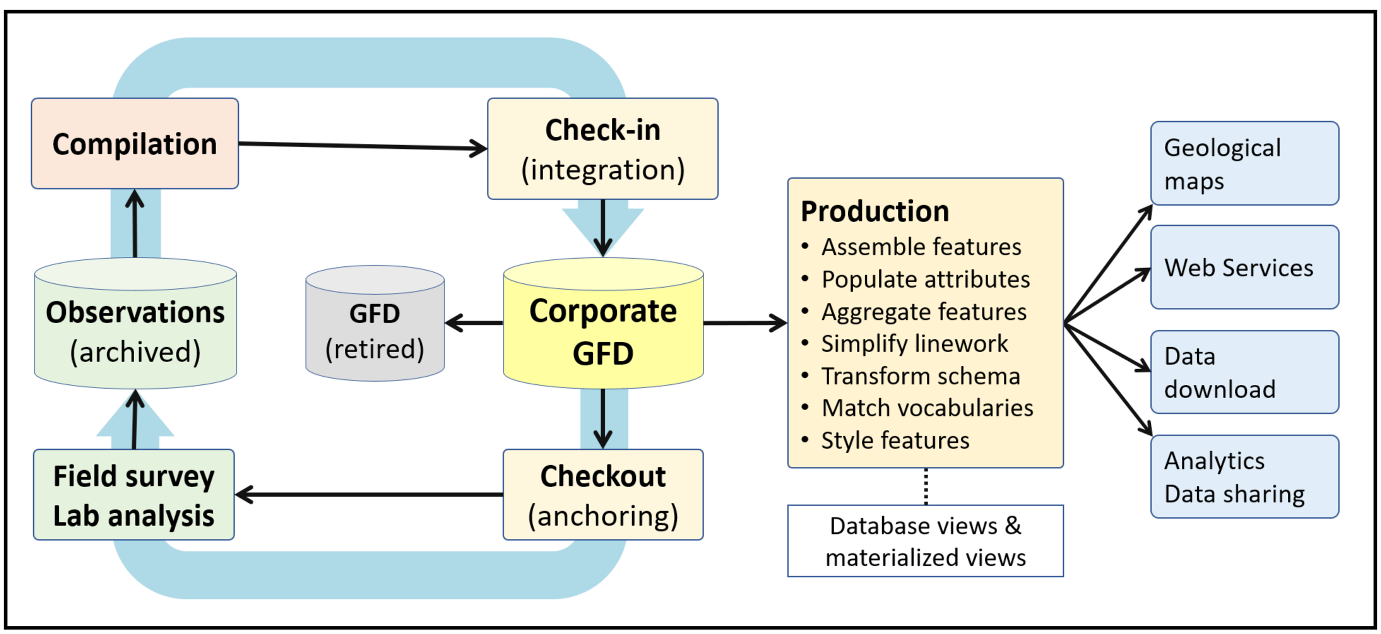

Fig. 19. GFD lifecycle and workflow, from data checkout to production.

and to support multiple levels of details (mapping scale and presentational scales). The GFD boundaries also include topographic features required to close bedrock polygons at the provincial border, and at the edges of areas covered by water bodies or glaciers.

To start a new mapping project, the GFD data checkout application extracts a data package as the base for our geologists to update. The essential part of the application in a SQL statement is the single pass to create two buffers, one to select all the features intersecting the limit of mapping, and another to tag feature components as anchor lines, rode lines, and for revision (Fig. 20). The script uses bedrock polygons (built from GFD feature components) to extract the area to be updated (extended beyond the limit of mapping), and to create a buffer to select all the affected feature components and tag them with a unique mapping project identifier (Fig. 21a). The selected feature components in a buffer of the extended area boundaries are tagged as 'anchorline', and those intersecting the buffer of the extended area boundaries are tagged as 'rode line', and the rest are tagged as 'for revision' (Fig. 21b).

```
SELECT ST_Buffer(ST_BuildArea(ST_ExteriorRing(ST_Union(a.geom))), 1)
    geom_buff_aoi,ST_Buffer(ST_ExteriorRing(ST_Union(a.geom)), 1)
    geom_buff_anchorline
FROM (SELECT a.geom FROM mv_bedrock_poly a LEFT JOIN mp_areas_poly b ON
    ST_Intersects(a.geom, b.geom)
WHERE b.mp_id = 'my_map_project_id') AS foo;
```

Fig. 20. SQL snippet for GFD data checkout.

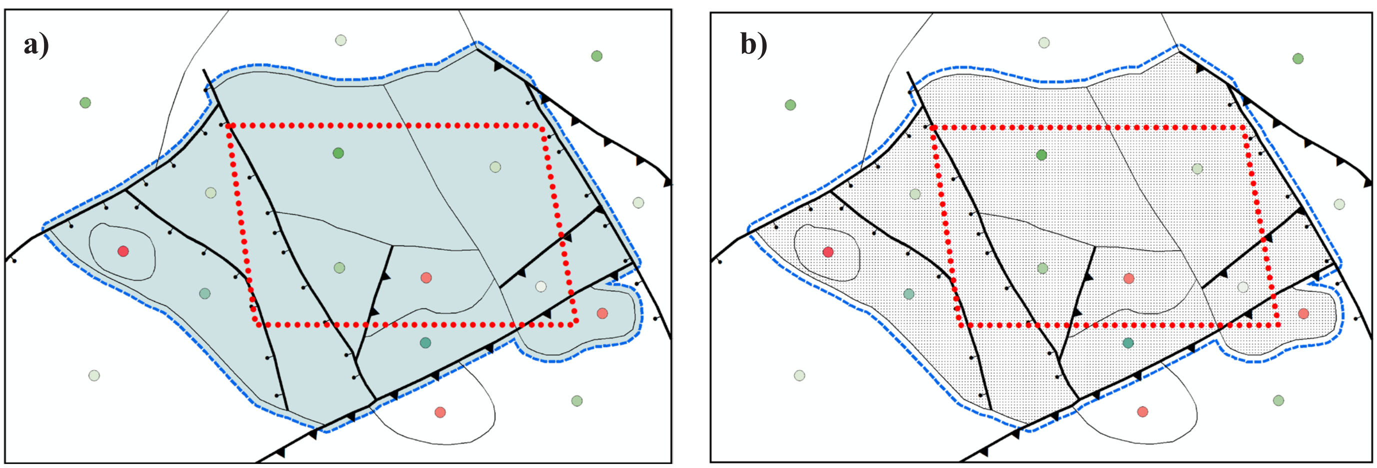

Fig. 21. Using buffer to select GFD feature components for a new mapping project shown as dotted line in red for limit of mapping. **a)** Creating a buffer [geom_buff_aoi] to tag feature components that should be included in the mapping project area; **b)** Creating a buffer [geom_buff_anchorline] to tag anchor lines, anchor points, rode lines, and the rest for revision.

```
-- Trigger: track change to boundary
CREATE TRIGGER track_change_bndy
  BEFORE INSERT OR DELETE OR UPDATE ON
gfd_bndy_lines
  FOR EACH ROW
  EXECUTE PROCEDURE track_change();
-- Trigger: track change to centroid
CREATE TRIGGER track_change_centroid
  BEFORE INSERT OR DELETE OR UPDATE ON
gfd_centroids
  FOR EACH ROW
  EXECUTE PROCEDURE track_change();
-- Function: track_changes()
CREATE FUNCTION track_changes()
  RETURNS trigger
  LANGUAGE 'plpgsql'
  COST 100
  VOLATILE NOT LEAKPROOF
AS $BODY$
DECLARE ...;
......
```

Fig. 22. Database trigger function snippet to track changes and validation.

We can choose desktop GIS or a staging area in the BCGS corporate GFD database environment to carry out digital compilations from mapping projects and data validation. In the case of the corporate GFD database, we track changes with database trigger functions on revisioning actions such as *insert* (adding new feature components), *delete* (retiring deleted feature components), or *update* (modifying existing feature components). The trigger functions also track validation on data quality assurance (QA) and content standardization, including QA status (passed, failed, issues detected, review, and issue resolution). The history of revisioning and QA includes details such as what (insert, delete, update, or validate), when (time-stamp), who (database user name), and why (reasons for changes). The trigger functions can act before or after changes to the attributes or geometries for the GFD boundaries or centroids are applied (Fig. 22). As an example, to validate the

GFD geometries, a simple SQL statement can detect lines that are not noded properly to form polygons, or if centroids are missing or duplicated (Fig. 23).

After validation is complete and the updates are ready for integration, outdated GFD feature components tagged as rode lines and for revision in the corporate GFD database are retired to an archival database (Fig. 24). The database trigger function tracking changes also can handle this in response to the *delete* action. The next step is to upload the updated feature components into the corporate GFD database (Fig. 24).

The GFD model is designed with anchor lines beyond the limit of mapping. If the anchor line is split to accommodate a new boundary (Fig. 17b), it can be handled in two ways. The first changes the anchoring tag (e.g., from 'anchorline'

```
SELECT pid FROM my_bedrock_poly
 WHERE gid NOT IN (SELECT a.gid
  FROM my_bedrock_poly a, my_bedrock_centroids_sp b
WHERE ST_Contains(a.the_geom, b.the_geom)
GROUP BY a.gid HAVING COUNT(a.gid) = 1 ORDER BY a.gid) ORDER BY gid;
```

Fig. 23. SQL statement snippet to test matching of centroids and polygons to ensure lines are noded properly to form the correct number of bedrock polygons.

```
-- Retire outdated feature components in the corporate GFD database
INSERT INTO bc_gfd_bndy_sp_retired (
            lid, f_class, f_type, f_conf, f_name, mp_id, ckout_tag, ...
       FROM bc_gfd_bndy_sp
      WHERE mp_id = 'my_mapping_project_id'
            AND (ckout_tag = 'rodeline' OR ckout_tag = 'revision');

DELETE FROM bc_gfd_bndy_sp
      WHERE mp_id = 'my_mapping_project_id'
            AND (ckout_tag = 'rodeline' OR ckout_tag = 'revision');

INSERT INTO bc_gfd_centroid_sp_retired (
            pid, map_unit, fm, age_max, age_min, mp_id, ckout_tag, ...
       FROM bc_gfd_centroid_sp
      WHERE mp_id = 'my_mapping_project_id' AND ckout_tag = 'revision');

DELETE FROM bc_gfd_centroid_sp
      WHERE mp_id = 'my_mapping_project_id' AND ckout_tag = 'revision');

-- Upload updates from new mapping into the corporate GFD database
INSERT INTO bc_gfd_bndy_sp (
            lid, f_class, f_type, f_conf, f_name, mp_id, ckout_tag, ...
       FROM my_gfd_bndy_sp
      WHERE ckout_tag <> 'anchorlin');

INSERT INTO bc_gfd_centroid_sp (
            pid, map_unit, fm, age_max, age_min, mp_id, ckout_tag, ...
       FROM my_gfd_centroid_sp
      WHERE ckout_tag <> 'anchorline');
```

Fig. 24. SQL statement to retire outdated feature components in the corporate GFD database and upload updates from a new mapping project.

to ‘anchorline_split’) in the updates to identify the cases. The second uses SQL script to test if any of the rode lines from the updates intersect anchor lines in the corporate GFD database without anchor points (Fig. 25). After the cases are detected, a GIS tool or spatial function in the database (e.g., PostGIS ST_Split) can be used to split anchor lines in the corporate GFD database to add anchor points.

In an ideal situation, all the rode lines in the updates are hooked properly to anchor points in the corporate GFD database. Coordinate drifting of updated feature components is usually eliminated from rounding coordinates to an appropriate unit of precision. If drifting exceeded the minimum unit of precision but was still within acceptable range (e.g., within 5 metres for a mapping scale at 1:50,000), SQL scripts can snap the rode lines from the updates to the anchor points in the corporate GFD database. An example SQL script that can be used to snap the start ends of the rode lines to the start ends of anchor lines is shown in Figure 26. Similar scripts can be created to handle other combination of line ends, or to use a different approach in snapping the line ends. The SQL scripts in Figure 25 and Figure 23 can be used for final testing to ensure that all rode lines are hooked with anchor points and that the GFD lines can form the correct bedrock polygons with matching centroids.

In map production, we use database views and materialized views to customize maps or data products with choices on levels of detail and feature types. As an example, we used a database view to simplify the GFD boundaries by using a tolerance of 5 metres (Fig. 27) and created a materialized view (Fig. 28)

```
-- Test if any rode lines in updates intersect anchor lines
-- in the corporate GFD database without anchor points
SELECT b.lid
  FROM bc_gfd_bndy_sp a, my_gfd_bndy_sp b
 WHERE a.ckout_tag = 'anchorline' AND a.mp_id = 'my_mapping_project_id'
   AND b.ckout_tag = 'rodeline'
   AND ST_Intersects(ST_Buffer(a.geom, 10), b.geom)
   AND (ST_Distance(ST_StartPoint(a.geom), ST_StartPoint(b.geom)) > 10
   AND ST_Distance(ST_StartPoint(a.geom), ST_EndPoint(b.geom)) > 10
   AND ST_Distance(ST_EndPoint(a.geom), ST_StartPoint(b.geom)) > 10
   AND ST_Distance(ST_EndPoint(a.geom), ST_EndPoint(b.geom)) > 10);
```

Fig. 25. SQL statement snippet to test if any rode lines in updates intersect anchor lines in the corporate GFD database without anchor points. A tolerance of 10 metres is used to search rode lines.

```
-- Snap rodeline.StartPoint to anchorline.StartPoint
UPDATE bc_gfd_bndy_sp a
SET geom = ST_SetPoint(a.geom, 0, ST_StartPoint(b.geom))
FROM bc_gfd_bndy_sp b
WHERE a.mp_id = 'my_mapping_project_id' AND a.ckout_tag = 'rodeline'
  AND b.mp_id = 'my_mapping_project_id' AND b.ckout_tag = 'anchorline'
  AND ST_Intersects(ST_Buffer(ST_StartPoint(b.geom),5),
      ST_StartPoint(a.geom))
  AND NOT ST_Equals(ST_StartPoint(b.geom), ST_StartPoint(a.geom));
```

Fig. 26. SQL statement example to snap the start ends of rode lines to the start ends of anchor lines within a tolerance of 5m.

```
-- Database view to simplify GFD boundaries
-- within a tolerance of 5 metres
CREATE OR REPLACE VIEW v_geo_bndy_simplified_5m AS
 SELECT gid, f_type, f_name, ...
        ST_SimplifyPreserveTopology(geom, 5) AS geom
   FROM bc_gfd_bndy_sp;
```

Fig. 27. Database view to simplify GFD boundaries within a tolerance of 5 m.

```sql
-- SQL Materialized View to form polygons from a database view
-- and populate bedrock attributes from centroids
CREATE MATERIALIZED VIEW mv_bedrock_poly AS
 SELECT a.gid,a.map_unit,a.unit_age,a.unit_name,a.rock_type,...d.geom
   FROM bc_gfd_centroid_sp a,
  (SELECT g.geom::geometry(Polygon,3005) AS geom FROM (SELECT
           ST_Dump(ST_Polygonize(v_geo_bndy_simplified_5m.geom))).geom AS geom
     FROM v_geo_bndy_simplified_5m
    WHERE v_geo_bndy_simplified_5m.f_type <> 'alternation'
      AND v_geo_bndy_simplified_5m.presentation_scales LIKE '%250000%') g) d
    WHERE bc_gfd_centroid_sp.presentation_scales LIKE '%250000%'
      AND bc_gfd_centroid_sp.rock_type <> 'alternation'
      AND ST_Contains(d.geom, a.geom) WITH DATA;
```

Fig. 28. Database materialized view to form bedrock polygons and populate attributes from GFD centroids.

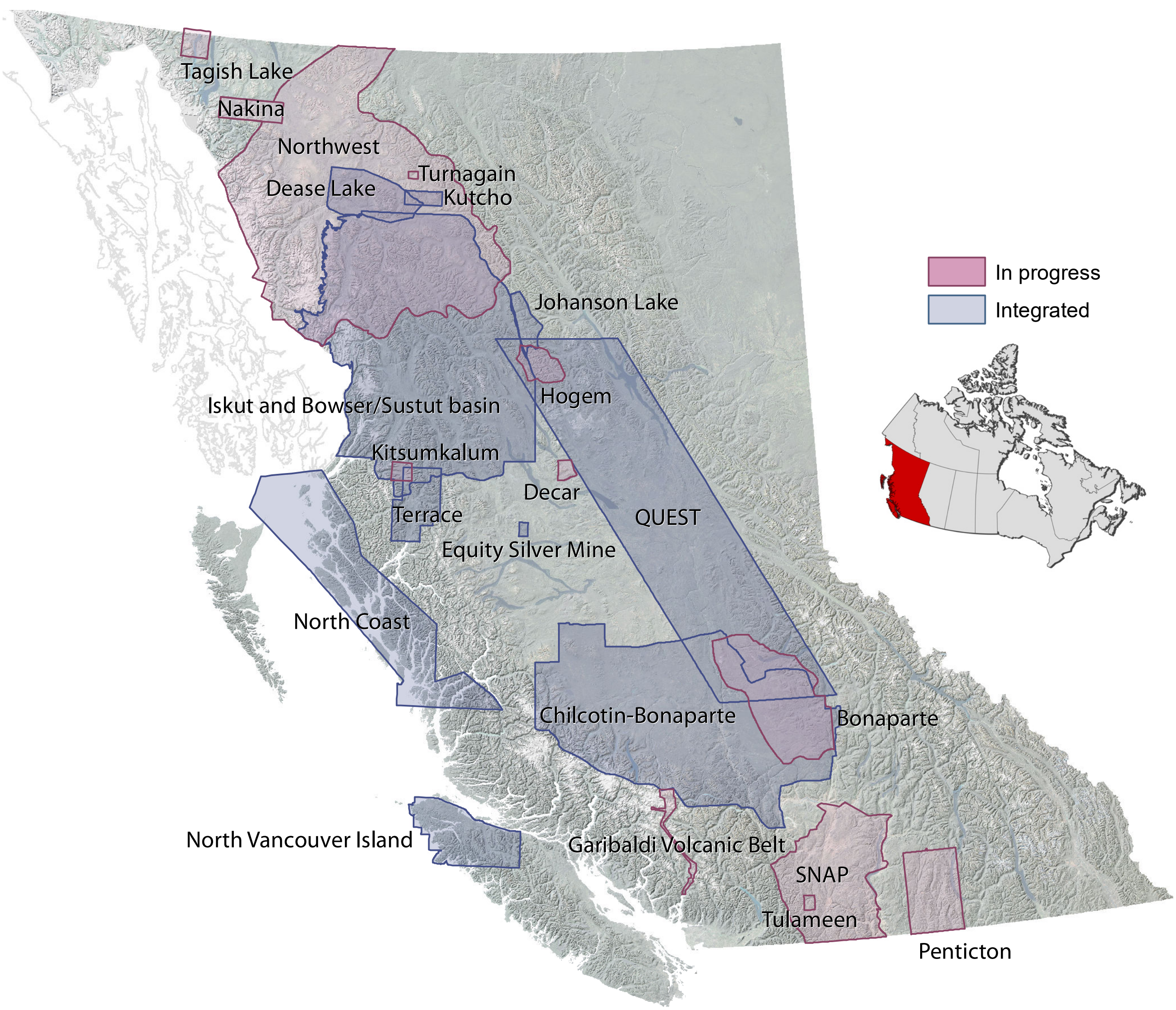

Fig. 29. Progress in regional compilation and map integration in British Columbia.

to form bedrock polygons (Fig. 18b) from the simplified lines and using only feature components tagged for presentation at a scale of 1:250,000, excluding those tagged as alteration (Fig. 17a). The database views and materialized views provides flexibility to create finished products without actual changes to the data in the GFD database; the database view generates a copy of simplified GFD boundaries only in the memory. At the BCGS, we need maps and digital products at multiple scales or level of detail to support many applications, such as rapid display of maps at multiple zoom levels on MapPlace 2, the BCGS geospatial web service (Cui et al., 2018b) or generalizing and aggregating features to balance data density in mineral potential modelling. This approach only requires refreshing the materialized view to update the maps or data products when the corporate GFD database has integrated new feature components.

We have been using the BCGS corporate GFD database to integrate data from new mapping projects to update the BC Digital Geology (Fig. 29) while continuing our efforts to gain efficiency through optimizing the process and improving the database applications. Our example is neither the only way to implement, nor the only choice of technology.

5. Conclusions

We developed the geospatial frame data (GFD) model to simplify digital map compilation and data integration. The model eliminates topological errors in edge matching, removes map boundary problems, and yields the flexibility to create custom maps and data products. The GFD model adds digital data compilation and integration stages to the geospatial data lifecycle. These intermediate stages processes raw observational data to construct the most primitive and detailed feature components that are digitally consistent and machine readable. Significantly, even though these stages may not be readily apparent in a finished map, they allow tracking back to original observations, thus enabling the data to be easily modified or repurposed into different final products. The GFD feature components become readily consumable source data to assemble finished data products with specified levels of detail and maps at multiple scales or for specific purposes. Using only lines and points, the GFD model can effectively represent both linear and polygonal feature components. By dispensing with polygons (except at the final map product stage), the GFD model eliminates gaps, overlaps and other topological errors caused by rounding coordinates, both in the GFD source data and in the resulting finished maps and data products. Using only lines to represent all linear and areal feature components, notorious 'map sheet boundary faults' disappear, and topological errors from editing and updating feature components at shared boundaries are easily reconciled. A data checkout process and anchoring mechanism reduce the risk of introducing new map boundary problems and simplifies map integration by eliminating tedious edge matching and associated errors. Without complex topological constraints, the GFD model can be easily implemented in a spatial database with applications for data checkout, anchoring, validation, and map integration. Application database 'views' and 'materialized views' can be developed to operate on the GFD feature components and automatically create custom maps and data products.

Acknowledgements

This research is funded by the British Columbia Geological Survey, British Columbia Ministry of Energy, Mines and Low Carbon Innovation. The work benefitted from discussions with David Skea and Martin Davis on topological errors associated with unit of precision. I thank Adrian Hickin, JoAnne Nelson, Paul Schiarizza, Larry Diakow, Deanna Miller, and other colleagues at the British Columbia Geological Survey for their encouragement and support in the implementation and practices. The manuscript was improved from reviews by David Soller (United States Geological Survey), Harvey Thorleifson (Minnesota Geological Survey), and Evan Orovan (British Columbia Geological Survey).

References cited

Aronoff, S., 1989. Geographic Information Systems: A Management Perspective, WDL Publications, 294pp. https://doi.org/10.1080/10106048909354237

Chrisman, N, 2006. Charting the Unknown: How Computer Mapping at Harvard Became GIS. ESRI Press, Redlands, CA. 228pp.

Clementini, E., Di Felice, P., and van Oosterom, P., 1993. A small set of formal topological relationships suitable for end-user interaction, In: Abel, D., Ooi, B.C. (Eds.) SSD 1993: Advances in Spatial Databases. Lecture Notes in Computer Science, Vol. 692, pp. 277-295, Springer: Berlin, Heidelberg https://doi.org/10.1007/3-540-56869-7_16

Clementini, E., Sharma, J., and Egenhofer, M.J., 1994. Modelling topological spatial relations: Strategies for query processing. Computers & Graphics 18, 815–822. https://doi.org/10.1016/0097-8493(94)90007-8

Cui, Y., Hickin, A.S., Schiarizza, P., Diakow, L.J., Miller, D., Nixon, G.T., Nelson, J.L., and Ferri, F., 2018a. Methods to update the digital geology of British Columbia and synopses of recently integrated mapping programs. In: Geological Fieldwork 2017, British Columbia Ministry of Energy, Mines and Petroleum Resources, British Columbia Geological Survey Paper 2018-1, pp. 197-215.

Cui, Y., Zhao, S., Fortin, G., Meredith-Jones, S., and Jones, L.D., 2018b. Development of MapPlace 2 geospatial web service to disseminate geoscience in British Columbia. In: Geological Fieldwork 2017, British Columbia Ministry of Energy, Mines and Petroleum Resources, British Columbia Geological Survey Paper 2018-1, pp. 183-195.

Egenhofer, M., and Herring, J., 1990. A mathematical framework for the definition of topological relationships. In: Brassel, K., Kishimoto, H. (Eds.), Proceedings, Fourth International Symposium on Spatial Data Handling, Vol 2. International geographical union, Zurich, Switzerland, pp. 803-813.

Egenhofer, M., and Franzosa, R., 1991. Point-set topological spatial relations. International Journal of Geographical Information Systems 5, 161-174. https://doi.org/10.1080/02693799108927841

Esri, 2010. ArcGIS Geodatabase Topology Rules. https://desktop.arcgis.com/en/arcmap/10.3/manage-data/editing-topology/pdf/topology_rules_poster.pdf (last accessed 2021-09-01)

Hurst, J., Mahoney, M.S., Taylor, N.H., Ross, D.T., and Fano, R.M., 1989a. Retrospectives I: the early years in computer graphics at MIT, Lincoln Lab, and Harvard. SIGGRAPH (Panels), pp. 19-38. https://doi.org/10.1145/77277.77279

Hurst, J., Mahoney, M.S., Gilmore, J.T., Roberts, L.G., and Forrest, A.R., 1989b. Retrospectives II: the early years in computer graphics at MIT, Lincoln Lab, and Harvard. SIGGRAPH (Panels), pp. 39-73. https://doi.org/10.1145/77277.77280

Massey, N.W.D., MacIntyre, D.G., Desjardins, P.J., and Cooney, R.T., 2005. Digital geology map of British Columbia. British Columbia Ministry Energy, Mines, Petroleum Resources, British Columbia Geological Survey Open File 2005-02, scale 1:250,000.

Martinez-Llario, J., Coll, E., Núñez-Andrés, M., and Femenia-Ribera, C., 2017. Rule-based topology system for spatial databases to validate complex geographic datasets. Computers & Geosciences, 103, 122 -132. https://doi.org/10.1016/j.cageo.2017.03.013

Open Geospatial Consortium, 1999. OpenGIS® Simple Features Specification for SQL, Revision 1.1; Open Geospatial Consortium, Inc. OpenGIS® Project Document 99-049, Release May 5, 1999, 78p.

Peucker, T.K., and Chrisman, N., 1975. Cartographic Data Structures. The American Cartographer 2(1), 55-69. https://doi.org/10.1559/152304075784447289

Ross, D.T., 1989. Oral history interview with Douglas T. Ross. Charles Babbage Institute. Retrieved from the University of Minnesota Digital Conservancy, 33p. http://hdl.handle.net/11299/107611

Si Alhir, S., 1998. UML in A Nutshell. O'Reilley Associates, Inc., 296p.

Theobald, D.M., 2001. Topology Revisited: Representing Spatial Relations. International Journal of Geographical Information Science, 15, 689-705. https://doi.org/10.1080/13658810110074519

van Oosterom, P., Stoter, J., Quak, W., and Zlatanova, S., 2002. The Balance Between Geometry and Topology. In: Richardson, D.E., van Oosterom, P. (Eds.) Advances in Spatial Data Handling. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-56094-1_16

Wahl, R.R., 2004. The use of topology on geological maps. In: Soller, D.R. (Ed.) USGS Digital Mapping Techniques Workshop Proceedings. United States Geological Survey Open-File Report 2004-1451, 6p.

BRITISH COLUMBIA

Ministry of
Energy, Mines and
Low Carbon Innovation

BRITISH COLUMBIA
est. 1895
GEOLOGICAL SURVEY