Cadastral System Modernization: the Technology and Business Requirements that Drive the Next Wave of Disruption

Terry BRINKMAN, USA

Key words: Cadastral systems, Parcel Fabric, Service Oriented Architecture, Quality Validation, 4D Cadastre, Branch Versioning

SUMMARY

The business requirements and expectations of a modern cadastral system can be tied directly to technological trends. People expect to be able to access live cadastral data from any device at any time and be able to trust it. Users should be able to access a certain set of capabilities based on their role or group association and know that any edits they perform are tracked and can be validated against a pre-configured set of business rules that ensures the data quality. Working in a distributed environment using service-oriented architecture (SOA) and guaranteeing performance and scalability requires adoption of latest protocols, RESTful stateless design, multithreading programming model and the use of GPU for 3D rendering. Organization should be able to deploy it on their infrastructure (on premise) or on the cloud with the option for virtualization. Mobile clients for field data collection should be able to perform edits while disconnected from the network and sync their edits when ready. Users should be able to view cadastral data at any historical moment in time both in 2D and 3D. The paper describes the requirements and the cutting-edge technology to support it.

Cadastral System Modernization: the Technology and Business Requirements that Drive the Next Wave of Disruption

Terry BRINKMAN, USA

1. BUSINESS REQUIREMENTS

The phrase ‘Location, location, location’ is often used to determine the desirability of a property. Software developments analogous phrase can be ‘Requirements, requirements, requirements’. A great effort is invested in collecting business requirements. Collecting business requirements requires subject matter experts that can not only understand the current workflows but also innovate and disturb the market by finding more efficient workflows. Esri R&D (Research and Development) activities include direct engagement with customers, business partners, consultants and reading professional material. Only once the requirements are well understood as well as the business value to the customer, can it move into the design phase.

Another great source of information is the ideas website, which allows individuals from the community to participate and either submit their own ideas or vote on existing ideas.

2. SOFTWARE DESIGN

A few decades ago software design used mostly UML ‘strawman’ and class diagrams. While some UML diagrams can be useful from time to time, it is more convenient and realistic to use a storyboards instead. A visual storyboards uses proposed interface in the same way (UI & UX) that the user will interact with it and uses the same context and environment. Storyboards are easy to create and can be easily modified based on design reviews until the final design is accepted. The design is led by experience subject matter experts and include experts from other teams such as the User Interface and User Experience team. Additional experts are pulled in as needed.

3. AGILE SOFTWARE DEVELOPMENT

Agile software development is comprised of a few approaches to software development that promote self-organized and cross-functional teams. With agile development the planning is adaptive to future changes. It is acceptable to deliver early and continue to improve over the rapid release cycles, rather than deliver the wrong things.

The agile development manifesto favors:

Individuals and interactions over processes and tools.
Working software over comprehensive documentation.
Customer collaboration over contract negotiation.
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left "more."

The agile software development cycle for an issue does not stop with design and implementation. For an issue to be done it has also to be manually and thoroughly tested, documented as part of the help documentation and automated tests are created to catch any regressions in functionality and performance.

All software development adheres to strict coding standards and code review process. Such a process guarantees that the code metes quality standards as well as that it can be maintained and extended by any developer.

4. SERVICE ORIENTED ARCHITECTURE

Services are re-usable. That is because the services provide or requested are complete and self-contained. Services are also platform independent and can be use by any type of client, scalable and reliable. With services cadastral data can be accessed and modified from any type of client in real time: desktop, mobile and web. Using user identity coupled with latest security protocols such as TLS 1.2 (Transport Layer Security), certain capabilities can be limited to specific groups.
The use of modern, light weight compressed formats such as JSON and Protocol Buffers from google, ensure fast performance and minimal transfer and parsing times. Services expose capabilities using modern REST API in a stateless fashion – the server does not store any state about the client session on the server side. To ensure top performance the communication is kept to the required minimum and optimizations are performed both on the server and client side. Such example of optimization is client-side caching of maps and layers.

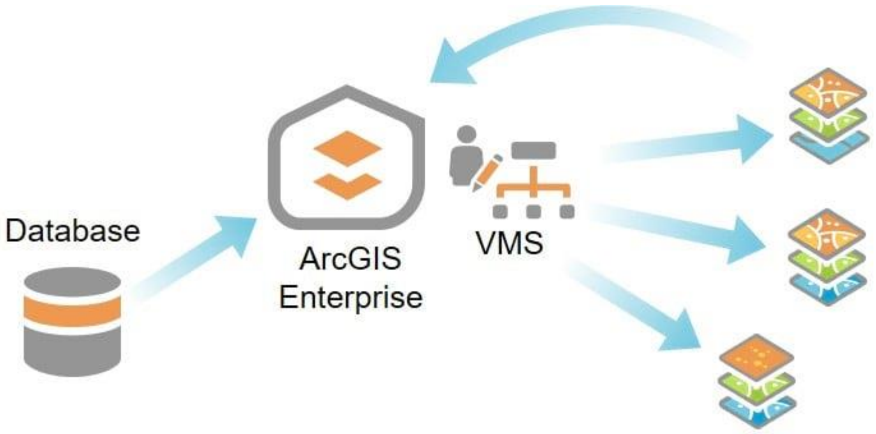
Another advantage of services is the ability to run heavy memory and CPU processes on the server from thin clients. Instead of running the process on a mobile device or a web browser, they are executed on the server, where they can leverage both the server resources as well as proximity to the actual data.

Instead of running long processes and having the server time out, some processes can be executed in an asynchronous fashion as well as being queued to preserve server resources.

5. BRANCH VERSIONING

With versioning, multiple people can edit simultaneously on a single feature class in a highly isolated fashion, without creating copies of the data. For each cadastral record a matching version can be created. A version is not a physical copy of the data but a logical one and it can allow the cadastral editor to perform each transaction in a dedicated version and then transfer the version to the person performing the final quality assurance (QA). If it passes QA, the changes can be pushed to the master version called ‘default’.

With the new Branch Versioning that uses services, all operations: inserts, updates as well as ‘deletes’ are a fast insert operation that are time stamped with the user’s identity.



A few advantages of such approach:

- Performance
- Each operation is tracked with the user’s identity.
- Undo of an edit operation is moving to a past moment.
- Ability to move to any given moment in time which provides a true 4D Cadastre capability.

6. HIGH LEVEL SCRIPTING LANGUAGES AND THE ATTRIBUTE RULES FRAMEWORK

Many GIS operations, from symbolizing a layer, trough labeling and popups, to calculating a field require an expression. While such expressions can utilize languages like Python in the desktop environment, they cannot be installed in the mobile and web environment: they are too big and pose a security risk. Arcade is a secure expression language that can run on any device. With Arcade an expression is defined once and works across all client types. Arcade is a GIS centric expression language that provides spatial operators as well as access to the feature's geometry.

Attribute Rules is a new framework that uses Arcade. Attribute rules can be configured with the organizations business logic to perform the following:

- **Calculation** – calculate attributes in a specific order to improve efficiency. For example – calculate a land description string by intersecting a feature with other administrative boundaries layers and combine it with the parcel identifier attribute value.
- **Constraint** – prevent bad data from being created by evaluating an attribute value and providing a configured error message. For example: prevent a parcel from being created if the parcel name contains illegal characters.
- **Validation** – existing features can be evaluated against attribute rules. Any violation creates an error feature. For example: compare the legal area to the shape areas (geometry) using a given tolerance.

7. MODERN – ARCGIS PRO AS PREMIUM CLIENT

ArcGIS Pro is a modern 64-bit, multithreaded desktop application that takes advantage of all your computing resources. You can use ArcGIS Pro to visualize 2D maps side by side with 3D local scenes or global views. The easy-to-use ribbon interface is contextual, and the application is localized to many languages, including right to left languages (Arabic and Hebrew). ArcGIS Pro uses a project – a project can contain multiple maps and layouts as well as other common resources.
ArcGIS Pro is tightly integrated with the entire ArcGIS platform, making it easier to share and consume content to your organization, groups and the public.
All the parcel fabric capabilities are part of ArcGIS Pro and come to live when using a local file geodatabase. When using an enterprise deployment and services, ArcGIS Pro delegates the request to the appropriate REST API on the server instead.
ArcGIS Pro can be deployed in a virtual environment on premise as well as on the cloud and is the premium client for the professional users.

8. WORKS 'OUT OF THE BOX'

A few conscious decisions were made to make the parcel fabric technology work 'out of the box' in ArcGIS Pro:

- Great layer defaults – when adding a parcel fabric to a new map the layers use dedicated parcel layers that have good multiscale symbology, smart labeling, appropriate feature templates and much more. These defaults can be used 'as is' or modified to meet specific business needs.
- Tasks – tasks are user capture repeatable workflows and guide the user step by step, using the business language, to make them productive even if they are

new to ArcGIS Pro. Common parcel editing workflows are shipped with the software but can be easily extended and configured without any coding.

- No need to download any add-in or custom code – all the functionality comes with ArcGIS pro and there is no need to download any Add-In. Add-Ins are custom code and require to be compiled with each release or become stagnant on an old release.
- Can use any editing tool – with the new parcel fabric, the use of simple feature classes allows any editing tool to be used as needed. Instead of trying to prevent 'bad edits', any edit can be evaluated against the business rules at any time.
- Core technology – when using ArcGIS Pro there is no need to purchase additional ArcGIS Pro extension license. It is a core technology that is included with ArcGIS Pro.

9. PERFORMANCE AND SCALABILITY

Cadastral systems need to be able to scale from a few hundred parcels to a few hundred million parcels while providing the same user experience and performance. Such scalability is achieved by using proven scalable DBMS products such as Oracle, MS SQL Server and PostgreSQL. ArcGIS Enterprise can also be scaled up by adding additional resources as well as additional servers for load balancing and high availability (fail-over) technology.

Daily automated performance test are performed to catch any regressions in performance and to make sure the SLA criteria are always met. The performance test harness technology is also used to create virtual users. With a given hardware configuration the number of virtual users is increased until it is saturated.
The next phase is to move to Kubernetes – an open source container orchestration system for automating application deployment, scaling and management in a cloud computing environment and the use of micro-services.

10. CONFIGURATION OR CUSTOMIZATION?

A software that can be configured, allows the user to control the appropriate settings without the need to hire a developer. A customized system on the other hand, relies on custom code that is built on top of the core software. While custom code allows for additional automation and tailor-made user interface, it often requires to be recompiled and tested with each release. Usually, organizations that chose to heavily customize their system, find it difficult to stay current with technology and become stagnant on older releases, sometimes unsupported release and a high maintenance cost. Custom systems also require custom manuals, custom training and custom technical support. For that reason, a better approach is to discover and expose configuration capabilities for customers to leverage using a user interface that is fully supported.

11. AUTOMATION & SYSTEM INTEGRATION

Automation can be achieved using Geoprocessing models, Python script tools, .Net Pro SDK and REST APIs.
Each type of API can be used for different types of automation and integration with business systems. When creating and exposing an API it is important to understand the use cases in which they might be used: digital submission, automation, validation, 3D Data flow, analysis etc.

12. CONCLUSIONS

Building a state of the art cadastral system requires a team of experienced subject matter experts that works closely with end users to design and implement the right thing. The use of modern service oriented architecture provides improved security, scalability and integration with business systems. Software quality is kept to high standards using automated tests to maintain high performance and catch any regressions. Foundational frameworks such as Branch Versioning, Attribute Rules, Tasks and many others push the envelope and can be configured to meet specific business requirements. Using subject matter experts not only guarantees building the ‘right thing’ but also that it is “built right’ for specific cadastral use cases and with defaults that make it easy to adopt and deploy.
The use of Agile software development methodology and strict coding standards has many advantages in the short and the long term.

BIOGRAPHICAL NOTES

Terry Brinkman completed his Bachelor of Science in Computer Science (Honours), with a minor in GIS in 1992 @ Curtin University, Perth, Western Australia. He worked for Esri Australia 1993-1997. In 1997 he joined Esri Inc in Redlands, California, USA. Terry worked on Esri Survey technology starting in 2003. In 2005 he started working on Esri desktop Parcel Fabric solution as Lead Developer. In 2014 he started working on the current version of Parcel Fabric (SOA offering) and continued his role as Lead Developer.

CONTACTS

Terry Colin Brinkman
Esri
380 New York Street
Redlands
USA
Tel. +1 (909) 793 2853
Email: tbrinkman@esri.com
Web site: www.esri.com