

Single Detailed Database vs. Multiple  
Scale Databases for Map Production  
-&-  
Database Update vs. Map Update

*Ad-Hoc Discussion Group*

Seminar on On-Demand Mapping

Barcelona, September 23, 2000

International Cartographic Association

Commissions on Map Generalization & Map Production

# Discussion Topics

- Single Detailed Database vs. Multiple Scale Databases for Map Production
  - pros and cons
  - consequences for production steps
- Database Update vs. Map Update
  - Database  $\leftrightarrow$  Map update ?
  - implications of single vs. multiple databases
  - update propagation

All in agreement that single  
Database is more desirable.

Why ? ...

- Real time data acquisition technology allows us to gather data in high precision.  
“Truth” => “Appropriateness”
- Today’s technology allows us to gather extremely accurate data into (or associate with) a single database

# Why Single DB more desirable:

- Data Quality - when you have multiple databases always wonder - which one's right ?
  - Coverage - completeness
  - Logical consistency
  - Authority
- If you handle different product sets, would be better to update single database than multiple sets of data

# Disadvantages of Single Database?

- Only disadvantage of using single Database for mapping is that generalization takes time.
- The advantage of multiple databases is that since generalization is so time-intensive, you do it up front, don't have to do it each time.

# So Who's Doing it Now ?

- While we were all in agreement that a single database for map production would be desirable, no one in the room could say that they were currently doing this.
- All used multiple data sets for multiple products.
- The closest anyone came was having different scale range databases that could be used to support multiple products within each scale range.

# Software Being Used

<b>Vendor</b>	<b>Software</b>	<b>Database</b>
Laserscan	LAMPS2	Gothic
ESRI	Arcinfo	SDE, Geodatabase
Smallworld	OO Programming	RDBMS

# How far away is technology ?

- How far away are we from being able to create multiple map products out of a single Database ? 5-10-15-20 years ?
- No consensus - however, feeling is that using a single database for large-medium scale is OK, but for it to support small scale would be much more difficult.
  - Multiple scales - spec data for range of scales



# User Example: Multi-Scale Databases for Generalization

- NIMA in the 1990's had a 10K DB, wanted to use it to create 25K and 1M scale maps.
- Ended up creating 3 levels of database to support mapping needs
  - Level 1
  - Level 2
  - Level 3

# Database

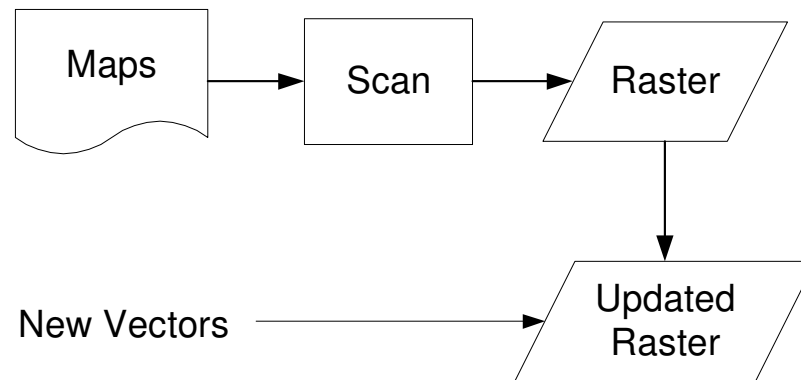
## Scales & Relationships

- 1:10K --> 1:1M or 1:500K --> 1:1M
- One database with knowledge of relationships

# User Example:

## Maps as Secondary Products

- Finished maps are scanned to raster
- New vectors are added (color tagged)
- These vectors are then processed by cartographers to create new, updated raster



# Discussion of Data Quality

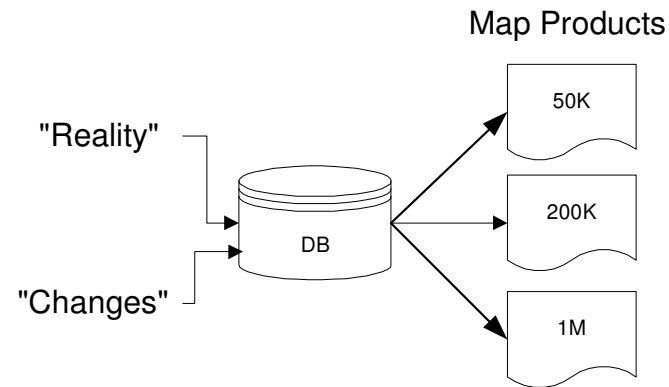
- Bottom -> Up      “Primary” Approach
- Top -> Down      “Secondary” Approach
- Use existing high quality data (1:50K), digitize analog to create model.
- In this context, the 1:50 scale data from existing manual product was deemed to be good enough - it was automated and serves as base data. Alternative would have been to gather all data from scratch using today’s highly accurate data gathering technology. Would have resulted in more accurate data, but at much higher (prohibitive?) cost. Deemed not necessary in this case, for this agency.

# Terminology: “Scale”

- Some discussion/ debate about appropriate terminology to describe data.
- “Scale” - does scale matter ? Instead...
  - Precision
  - Versioning
  - Resolution (of source data)
  - Accuracy
- Instead of thinking about ‘scale’ think about resolution.
- “Objectivity” - representation of reality

# The Dream: A Single Database & One-Time Update

- As you add more information to the database (to more accurately reflect reality, to incorporate changes) it automatically finds itself in all products & updates you derive from it.

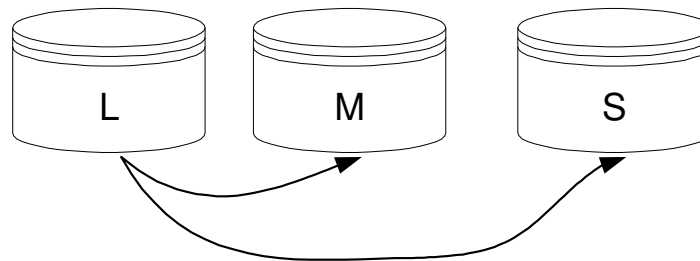


# Generalization & Updates

- Add a change -> notify targets
- Enrich database as much as possible
  - For your application needs
  - for Generalization
  - For you future update procedures

# Generalization Tools

- Make the data smart
- Make source and target talk to each other
- Links between different scale databases





# Data Volume, Update Frequency, Production Schedule

- Not just an issue of tools, also an issue of data volume
  - It's a lot of data to sift through
- Manual Work of Generalization
- Timing (schedule) or update frequency

# Object Oriented Technology

- Put enough information into object and build tools to represent it at any scale.
- Object Model
- Database Technology
- Object Views

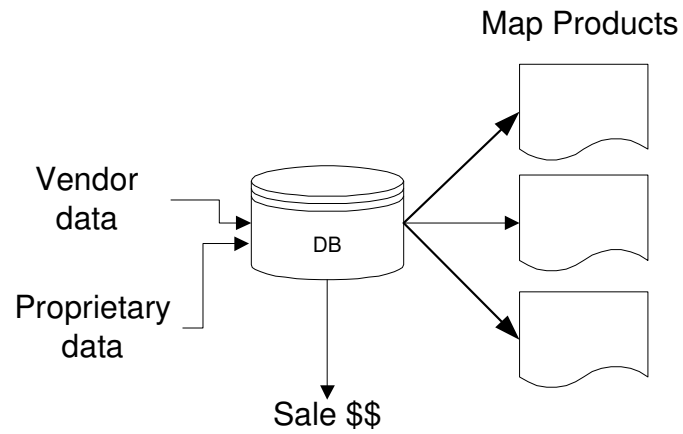
# Preserving changes, position in derived products

- Some way of knowing what products a change affects
  - object preserves displacement in all derived products
- vs.
- object determines correct context in all derived products
- Procedures, Transformation Matrices

# Middle Approach

- Users tell vendors: mechanisms, procedures for updating
- Model Automate Procedures
- Scale change big -- can you model ?
- Connections between features

# Update Scenario



What do we need to solve these  
problems ?

\$