Understanding and Managing Geodatabase Conflicts

David Reed – Dominion Power
Thomas Brown – ESRI, Redlands
Agenda

• Data model
• Workflow
• Topology – geometric networks
• Future Development Efforts…
  • Reconcile options
  • Network reconcile options
  • Column level filtering
  • Conflict resolution dialog
Data Model

- Relationships
  - Simple
  - Complex
    - 1:1
    - 1:M
    - M:M (attributed)

- Geometric Networks
  - In conjunction w/ relationships…
Workflow

- Version management
  - Flat version tree
  - Multi-level

- Affect on performance and increased conflicts
Version hierarchy

DEFAULT

CHILD

WorkOrder (1)

WorkOrder 1 is modified... inserts/updates/deletes
Version hierarchy

DEFAULT

CHILD

WorkOrder (1)

WorkOrder 1 is reconciled/posted to CHILD
Version hierarchy

DEFAULT

CHILD

WorkOrder (2)

WorkOrder 2 is modified… inserts/updates/deletes
Version hierarchy

DEFAULT

CHILD

WorkOrder (2)

CHILD is reconciled/posted to DEFAULT
Version hierarchy

DEFAULT

CHILD

WorkOrder (2)

CONFLICTS

WorkOrder 2 is reconciled/posted to CHILD
Why...

Inserts/Updates/Deletes

Reconcile and Post

WorkOrder (2)

CHILD

DEFAULT
Why…

Reconcile and Post

DEFAULT/CHILD

Inserts/Updates/Deletes

WorkOrder (2)

Reconcile and Post
Conflict Resolution

**Conflict:** Feature representation from the target version

**Edit:** Feature as it existed in the edit session

**Pre-Edit:** Feature as it existed prior to editing

<table>
<thead>
<tr>
<th>Property</th>
<th>Conflict</th>
<th>Edit</th>
<th>Pre-Edit</th>
</tr>
</thead>
<tbody>
<tr>
<td>FID</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>SHAPE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEQ_ID</td>
<td>33211070</td>
<td>33211070</td>
<td>33211070</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>14</td>
<td>6</td>
<td>14</td>
</tr>
<tr>
<td>PIPE_SIZE</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>ACC_NO</td>
<td>16-A</td>
<td>16-A</td>
<td>16-A</td>
</tr>
<tr>
<td>SEW_NO</td>
<td>38952</td>
<td>38952</td>
<td>38952</td>
</tr>
<tr>
<td>MATERIAL</td>
<td>STEEL</td>
<td>STEEL</td>
<td>STEEL</td>
</tr>
<tr>
<td>SEW_SHAPE</td>
<td>CIRC</td>
<td>CIRC</td>
<td>CIRC</td>
</tr>
<tr>
<td>HEIGHT</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>WIDTH</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>INST_YEAR</td>
<td>1982</td>
<td>1982</td>
<td>1982</td>
</tr>
<tr>
<td>DRAIN_AREA</td>
<td>S.BRANCH</td>
<td>S.BRANCH</td>
<td>S.BRANCH</td>
</tr>
<tr>
<td>SEP_COMB</td>
<td>SEPTIC</td>
<td>SEPTIC</td>
<td>SEPTIC</td>
</tr>
<tr>
<td>PUB_PRI</td>
<td>PUBLIC</td>
<td>PUBLIC</td>
<td>PUBLIC</td>
</tr>
</tbody>
</table>
Conflict Display

- Ability to specify conflict display environment

Conflict version: Display Color, Red

Edit version: Display Color, Green

Pre-edit version: Display Color, Yellow
Conflict Scenarios

- Simple Features
- Simple and Composite Relationships
- Geometric Networks
  - Common scenarios
Simple Conflicts

• Definition:
  – Any features updated or deleted in the version, which are also updated or deleted in the target version.
  – Conflicts occur even if it’s an attribute or spatial update
Simple Conflicts

- Two users each modify a gas meter’s attributes:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objectid</td>
<td>184723</td>
</tr>
<tr>
<td>Status</td>
<td>On</td>
</tr>
<tr>
<td>Install Date</td>
<td>June 12, 1999</td>
</tr>
<tr>
<td>Owner</td>
<td>Robert Allen</td>
</tr>
<tr>
<td>Address</td>
<td>9295 Beechnut</td>
</tr>
<tr>
<td>Type</td>
<td>Residential</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objectid</td>
<td>184723</td>
</tr>
<tr>
<td>Status</td>
<td>Off</td>
</tr>
<tr>
<td>Install Date</td>
<td>June 12, 1999</td>
</tr>
<tr>
<td>Owner</td>
<td>Robert Allen</td>
</tr>
<tr>
<td>Address</td>
<td>1917 Gregory</td>
</tr>
<tr>
<td>Type</td>
<td>Residential</td>
</tr>
</tbody>
</table>
Simple Relationships

- Simple relationships follow the same principles in regards to conflict detection as simple features.
  - Updating the Parent table in Version A and Updating the Child (related) table in Version B, **DOES NOT** generate a conflict
  - But, be aware: deleting the origin class feature’s row will set the foreign key to null on the destination class. This can create update/update conflicts
Composite Relationships

• Conflicts can arise when editing an origin class and destination class or when editing individual features which participate in the composite relationships
  – Examples…
Composite Relationships

• Subdivision/Parcel
  – Two users start editing the same subdivision and parcels which are contained within the subdivision in the DEFAULT version:

Parcels

Subdivision

All parcels with subid = 186, define the subdivision
Composite Relationships

• Subdivision/Parcel
  – User 1, splits a parcel and saves:

Parcels

Subdivision

All parcels with subid = 186, define the subdivision
Composite Relationships

- Subdivision/Parcel
  - User 2 updates an owner attribute on a different parcel:

  ![Parcels](image1)
  ![Subdivision](image2)

  All parcels with subid = 186, define the subdivision
Composite Relationships

• Subdivision/Parcel
  – User 2, saves and conflicts are detected:

All parcels with subid = 186, define the subdivision
Network Topology

• Common Scenario
  – 2 users start editing and insert new services to the same gas main (complex edge) in the SDE.DEFAULT version:
Network Topology

• Common Scenario
  – User 1, inserts a new service and saves:
Network Topology

• Common Scenario
  – The edits result in the following changes:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>Update</td>
</tr>
<tr>
<td>E2</td>
<td>Insert</td>
</tr>
<tr>
<td>J1</td>
<td>Update</td>
</tr>
<tr>
<td>J2</td>
<td>Update</td>
</tr>
<tr>
<td>J3</td>
<td>Insert</td>
</tr>
<tr>
<td>J4</td>
<td>Insert</td>
</tr>
</tbody>
</table>
Network Topology

• Common Scenario
  – User 2, inserts a new service:
Network Topology

- **Common Scenario**
  - The edits result in the following changes:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>Update</td>
</tr>
<tr>
<td>E3</td>
<td>Insert</td>
</tr>
<tr>
<td>J1</td>
<td>Update</td>
</tr>
<tr>
<td>J2</td>
<td>Update</td>
</tr>
<tr>
<td>J5</td>
<td>Insert</td>
</tr>
<tr>
<td>J6</td>
<td>Insert</td>
</tr>
</tbody>
</table>
Network Topology

• Common Scenario
  – User 2 saves, and conflicts are detected:

![Network Topology Diagram]

- J1
- E1
- J2
- J3
- E2
- J4
- E3
- J5
- J6
Network Topology

- **Common Scenario**
  - Reconcile result in the following conflicts:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Action</th>
<th>Feature</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>Update/Update</td>
<td>J1</td>
<td>Update/Update</td>
</tr>
<tr>
<td>E2</td>
<td>Insert</td>
<td>J2</td>
<td>Update/Update</td>
</tr>
<tr>
<td>E3</td>
<td>Insert</td>
<td>J3</td>
<td>Insert</td>
</tr>
<tr>
<td>J4</td>
<td>Insert</td>
<td>J5</td>
<td>Insert</td>
</tr>
<tr>
<td>J6</td>
<td>Insert</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Network Topology

• Conflict Propagation Rule…
  – Conflicts will always be propagated to the topologically related junction and no further
  – Reconciliation will ensure the successful union of topology and connectivity
Network Topology

• Complicated Scenario
  – 2 users start editing and insert new services to the same gas main (complex edge) in the SDE.DEFAULT version
  – But in this case, the complex edge contains a mid-span valve:
Network Topology

- Complicated Scenario
  - User 1, inserts a new service and saves:
Network Topology

- Complicated Scenario
  - The edits result in the following changes:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Action</th>
<th>Feature</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>Update</td>
<td>J1</td>
<td>Update</td>
</tr>
<tr>
<td>E2</td>
<td>Insert</td>
<td>J2</td>
<td>Update</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J3</td>
<td>No Change</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J4</td>
<td>Insert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J5</td>
<td>Insert</td>
</tr>
</tbody>
</table>
Network Topology

• Complicated Scenario
  – User 2, inserts a new service:
Network Topology

- Complicated Scenario
  - The edits result in the following changes:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Action</th>
<th>Feature</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>Update</td>
<td>J1</td>
<td>No Change</td>
</tr>
<tr>
<td>E2</td>
<td>Insert</td>
<td>J2</td>
<td>Update</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J3</td>
<td>Update</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J6</td>
<td>Insert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J7</td>
<td>Insert</td>
</tr>
</tbody>
</table>
Network Topology

• Complicated Scenario
  – User 2 saves, and conflicts are detected:
Network Topology

- Complicated Scenario
  - Reconcile result in the following conflicts:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Action</th>
<th>Feature</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>Update/Update</td>
<td>J1</td>
<td>Update</td>
</tr>
<tr>
<td>E2</td>
<td>Insert</td>
<td>J2</td>
<td>Update/Update</td>
</tr>
<tr>
<td>E3</td>
<td>Insert</td>
<td>J3</td>
<td>Update</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J4</td>
<td>Insert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J5</td>
<td>Insert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J6</td>
<td>Insert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J7</td>
<td>Insert</td>
</tr>
</tbody>
</table>
What to remember…

• How you model your data, will impact what is determined as a conflict
  – And performance…
  – Complexity of network topology
  – Number of relationships (simple or composite) between classes
Future Development

• Domain values
• Adding common ancestor version
  – Original object values
• Ability to traverse relationships (expand)
• User maintained checkbox
  – (is this valuable?)
• Ability to change conflict display symbol…
  (is this valuable?)
Future Development

- Column level conflict detection
  - Objectclass property
- Source version wins on reconcile...
  - Be aware: lost updates/inserts (data integrity)
- New reconcile behavior with geometric networks
  - No conflict escalation, fewer conflicts
  - Nothing is free… loss of connectivity
Conflict Resolution at Dominion

User Profiles

- 16 “Editors” (Short term transactions)
  3 Groups who attempt to geographically organize work

- 250 “Designers” (Long term transactions)
  Work is generally done in a specific geographic area with a work request polygon depicting the area where the edits are taking place
Geodatabase Versioning Configuration

• Editors
  • Public child of default (One off default)

• Designers
  • Create their first version as a public child of default (one off default)
  • Then they can make 2 children versions (alternatives) of the first version with the ability to post one of those children versions. They are essentially limited to 3 levels off of default. If they decide to post one of the children revisions, the posting service will delete the other versions in the family tree.

  • Copy an existing version with the ability to create alternatives mentioned above.
Conflict Resolution at Dominion

Geodatabase Statistics

- 8000+ Versions
- 1000+ Versions with conflicts
What kinds of conflicts?

- Mass attribute updates along a circuit such as feederid updates
- Edits to feature linked annotation features ie. Transformer > Transformer Annotation
- Edits to related feature classes ie. Pole > Transformer relationship
Conflict Resolution at Dominion

Impact of conflicts?

- Slower performing system.....decrease in performance by “pinning” records in the versioning (adds/deletes) tables

- Delay in posting of as built work

- Slower performing system.....
Conflict Resolution at Dominion

Automated Resolution process?

• Attempted to use conflict filters (limited success initially, but still evaluating the benefit)

• Always resolve conflicts with the edit version?? Loss of data is a big concern.
Conflict Resolution at Dominion

Manual Resolution process?

• Dedicate personnel to resolve conflicts full time to clear the backlog

• Determine which versions in the state lineage need to be reconciled the most and have conflicts were discovered by nightly/weekend batch reconcile

• Reconcile the versions, resolve the conflicts one by one.....
Conflict Resolution at Dominion

Tools to aid the manual Resolution process

• ASP.NET application for users to access information about the version

Display details about a version to assist in the conflict resolution effort

Enter a version name: 

TONY126.REV_5600660_2 -

Submit
Conflicts Resolution at Dominion

Version report for REV_5592942_4

<table>
<thead>
<tr>
<th>Version Name</th>
<th>Owner</th>
<th>Last Reconcile Status</th>
<th>Last Reconcile Date</th>
<th>WMIS Status</th>
<th>Version Is Orphaned</th>
<th>Resolution In Progress</th>
<th>Resolution Complete</th>
<th>Resolution User</th>
<th>Last Time Resolved</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFAULT</td>
<td>SDE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REV_5592942_1</td>
<td>GARY077</td>
<td>RECONCILED</td>
<td>07/06/2004 10:06</td>
<td>PCON</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REV_5592942_4</td>
<td>GARY077</td>
<td>CONFLICTS</td>
<td>07/06/2004 10:24</td>
<td>PCON</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>a2</td>
<td>10/04/2004 16:52</td>
</tr>
</tbody>
</table>

Your Enterprise id: [551003]

If you resolved conflicts on this version, please add some comments below:

Resolved with edit version to preserve designers edits.

Hit submit to signify that conflict resolution is in progress or resolution is complete

for Version: REV_5592942_4

Submit
Conflict Logging logs details about the conflict to a table.

<table>
<thead>
<tr>
<th>Conflict Date</th>
<th>Conflict Edit Version Name</th>
<th>Conflict Field Name</th>
<th>Conflict Edit Value</th>
<th>Conflict Parent Value</th>
<th>Table Name</th>
<th>Conflict Type</th>
<th>Conflict Nature</th>
<th>Conflict Parent Version Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>04/12/2004 01:29:11 PM</td>
<td>GARY077_REV_5592942_4</td>
<td>SHAPE</td>
<td></td>
<td></td>
<td>DOMINION.Misc_Conductor_Annotation</td>
<td>update-update</td>
<td>shape</td>
<td>GARY077_REV_5592942_1</td>
</tr>
<tr>
<td>04/12/2004 01:29:11 PM</td>
<td>GARY077_REV_5592942_4</td>
<td>SHAPE</td>
<td></td>
<td></td>
<td>DOMINION.Misc_Conductor_Annotation</td>
<td>update-update</td>
<td>shape</td>
<td>GARY077_REV_5592942_1</td>
</tr>
<tr>
<td>04/12/2004 01:29:11 PM</td>
<td>GARY077_REV_5592942_4</td>
<td>SHAPE</td>
<td></td>
<td></td>
<td>DOMINION.Misc_Conductor_Annotation</td>
<td>update-update</td>
<td>shape</td>
<td>GARY077_REV_5592942_1</td>
</tr>
<tr>
<td>04/12/2004 01:29:11 PM</td>
<td>GARY077_REV_5592942_4</td>
<td>SHAPE</td>
<td></td>
<td></td>
<td>DOMINION.Misc_Conductor_Annotation</td>
<td>update-update</td>
<td>shape</td>
<td>GARY077_REV_5592942_1</td>
</tr>
</tbody>
</table>
Conflict Resolution at Dominion

Lessons Learned

• Performance will be impacted big time as versions with conflicts accumulate in your system.

• Understand how complex data models with relationships and feature linked annotation can cloud the conflict resolution process.

• Scrutinize work flows that may increase the chance of conflicts.

• Make sure you have resources to resolve conflicts.

• Resolve those conflicts.

• Keep ESRI in the loop on your issues!!