

**Spatial Data  
Supply Chain:  
Delivering ROI**



# Reflections on Delivering ROI at Customer Sites

Simon Greener,  
SpatialDB Advisor  
Independent Oracle Spatial  
&  
Radius consultant

# Introduction

- This talk will:
  - Define Technical ROI
  - Outline experiences with delivering Technical ROI with Radius Topology at customer sites in Australia.
  - Describe ROI delivery at these sites
    - Issues and the solutions.
  - Conclude with comments on Radius Studio in light of the experiences with delivering Technical ROI for geospatial data management.

## ROI Sales and Marketing

- Software is sold with certain *assumptions* about how it will add value to an organisation.
- This ROI is *encapsulated* through marketing brochures and sales and marketing activities.
- But often the *initial* ROI that is extracted from a product can be quite different to the *final* or *target* ROI.

## But First – Who & What *Delivers* ROI?

- Fast, flexible technology *enables* the extraction of ROI from any organisation's spatial data and technology investment.
- But it is not, in itself, able to *deliver* ROI.
- The *agent(s)* through which the technology is delivered are the customer's staff and any external consultants.
- It is these two aspects that most affect product success and any measure of ROI.

# Technical ROI

- I define Technical ROI as the “space” in which ...
  - Vendor software
  - Consultants
  - Customer Staff
  - Databases
  - Business problems
- ... all meet to deliver multiple ROI outcomes.
- This is the space where: The “rubber hits the ground”

## Initial ROI Activities

- Initial technology take up requires everyone involved to “*walk*” before they can “*run*”.
- Process starts with Training and Mentoring.
- Followed by the identification of quick “wins” which cement technology acquisition and deliver a foundation from which more complicated outcomes can be built
  - “A house built on sand will not stand”.

# The Value of Training

- A critical foundation for ROI
- In general, technology training courses:
  - Are function-focused;
  - Use generic datasets;
  - Are hosted in generic training venues.
- But customers would prefer:
  - Solutions-focused courses (e.g., Radius Topology for water network management);
  - Use of their own datasets;
  - On-site training.

# Training Issues

- How can one match Expectation and Reality?
  - With difficulty!
- We know that successful training outcomes tend to be achieved when what is taught is applied, immediately afterwards, in a business context.
- But this assumes that:
  - The course content is **sufficient** to start the trainee on the road to making the knowledge *their* own.
    - The Vendor's responsibility.
  - The user is willing to accept that this is **their** responsibility
    - The Customer's responsibility
- Finally – Low expenditure by companies on staff training

## Mentoring – Solution and Issues

- One possible solution is for the trainer to provide “mentoring” services on customer data and rules formation in the days immediately after initial training.
  - This is the approach taken at the customer sites I have worked at.
- However, it is not a “silver bullet”.
- Issues with mentoring that affect ROI delivery:
  - Temptation for customer to use the mentor to develop tools and scripts (because customers are trying to do “more with less”).
  - Those assigned to do the training aren’t the ones that do the actual work (this has happened)!
  - Some implementation techniques depend on other factors:
    - Level of experience of parties (is there any substitute for experience?)
    - Solutions can have genesis in concepts learned through formal education.

# ROI Activity Essentials

- Regardless as to the background or methods used to facilitate and achieve them, good ROI delivery for data-centric technologies means concentrating on activities that:
  - Liberate staff from non-productive activities;
  - Reduce data processing and cleaning times (even for foreign data);
  - Discover, define and apply spatial data rules.
- All activities focus on:
  - data timeliness,
  - data definition,
  - data quality.
- The ultimate ROI we hope for is that the business decisions made from the information derived from our databases is the *best one possible*.

## Site 1 – Quality Data & Domain Knowledge

- Data Mature but explicit Rules Immature Site
  - High quality data
  - Strong business focus
- RT consultancy led to immediate concentration on rules discovery and encoding
  - Rules discovery done via GIS Manager meeting with Engineers (domain experts);
  - Initial focus was utility networks not external data (critical value-add);
  - Customer's clients surprised as to the number of rules created!

# Rules

- Lesson
  - Simple rules, even one's like “a end-cap can only connect to one pipe”, are assumed but not written down;
    - Simple but surprising!
  - Yet these are real rules that value-add databases
- A fully documented, managed and applied rule base is the best gift a GIS Manager can give to an organisation and those who follow after.

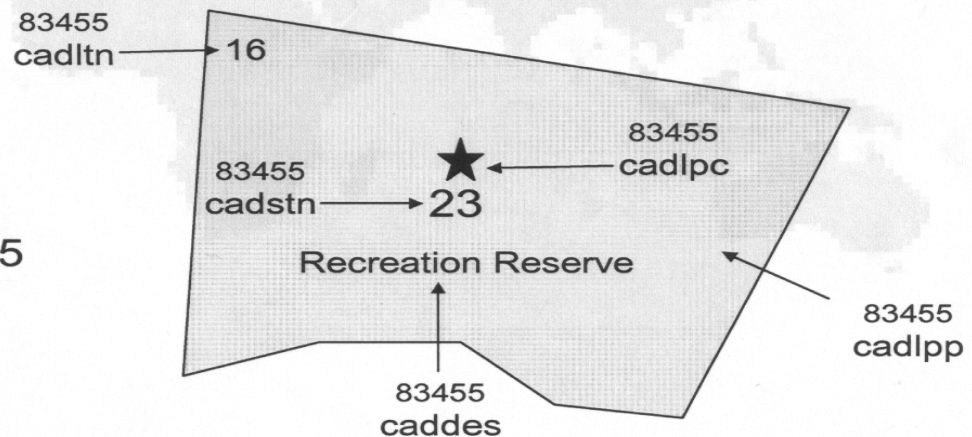
## Customer 2 - Migration

- Business Context
  - End ROI is same as customer one (same business)
  - But:
    - Customer migrating from CAD systems to Oracle Spatial
    - Land Parcels held as linestrings + points representing feature – called “Wireframe”
    - → Successful Data Re-Engineering first ROI

# Visually

## Feature Type: Land Parcel

Lot 16, No 23 Smith St Lara  
has landparcel IFC\_UID = 83455



- It is possible to have multiple points inside a parcel but only one has the actual parcel details associated with it.

## Customer 2 – Initial ROI

- Initial value-add of RT:
  - Convert the “Wireframe” to polygonal business features.
- Outcome:
  - Enable existing “Wireframe” oriented editing to continue as new polygon based applications (eg corporate visualisation) are deployed.
- Deliverable:
  - Batch conversion fast enough to enable overnight replication.
- Limitation:
  - Old hardware

## Customer 2 – Part 2

- Two Part Processing:
  - Structure and construct Sdo\_Geometry representation from “Wireframe” – line-2-poly conversion;
  - Attribution of polygons by integration with centroids.
- Approach:
  - Construct rules in RT for “Wireframe” and structure into planar manifold.
  - Use `Isl_topo_util.face_get_geometry()` to create potential polygons;
  - Search against parcel point data to validate that a face represents a parcel.
- Initial processing times (v12) – in excess of 24 hours

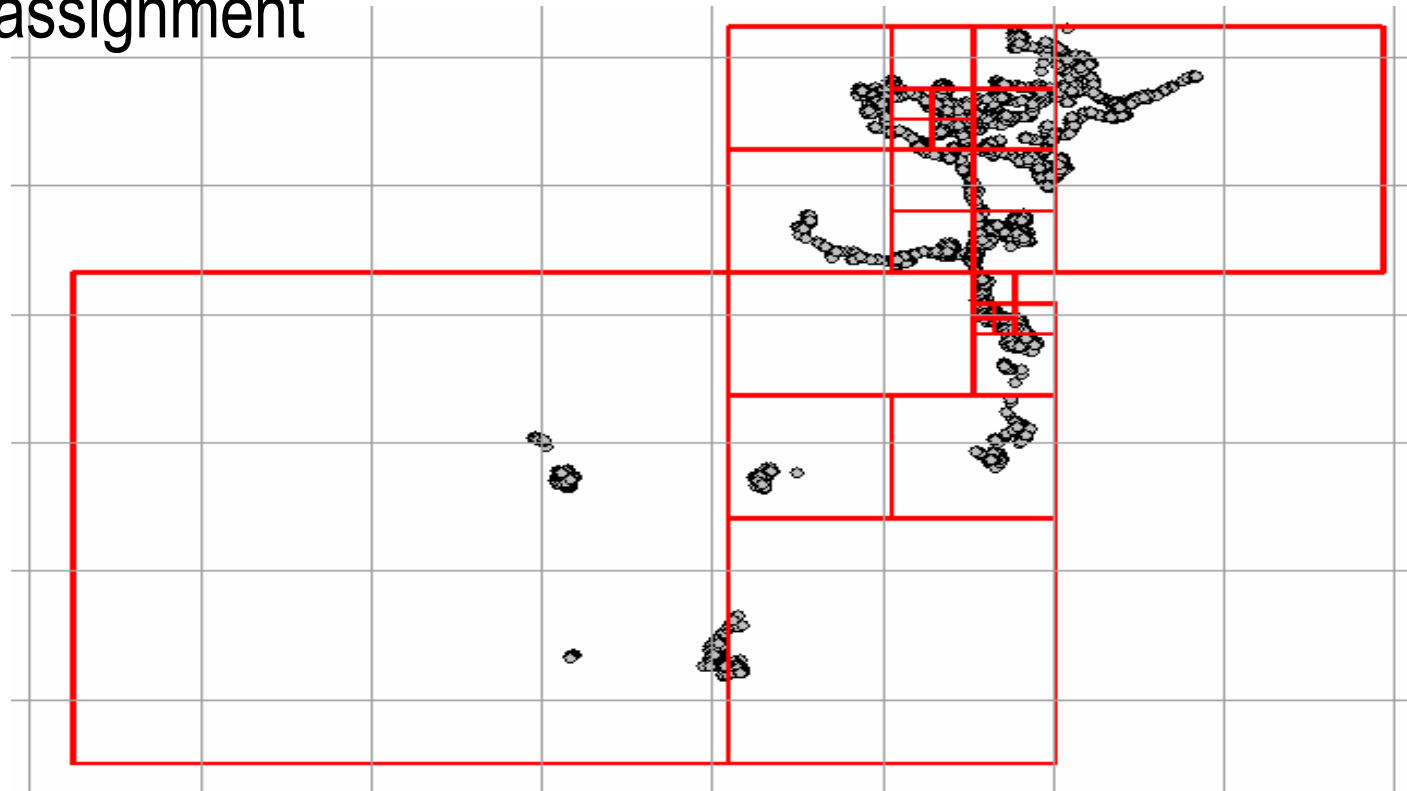
## Grid processing speed

- v22 distance processing saw big improvement but single cpu process
- Grid processing ROI difficult because of processing imbalance

Process	Timing
1	229,281 objs in 7 hrs 35 mins = 504 o/s
2	118,949 objs in 3 hrs 27 mins = 575 o/s

## Solution – Custom Quad Tree

- Needed more adaptive algorithm for grid construction and feature assignment



## Quad Tree - Results

Process	Quad Tree
1	172,526 objs in 6 hrs 15 mins = 460 o/s
2	170,625 objs in 6 hrs 8 mins = 464 o/s

- Slower object structuring speed probably down to server load

## Issue 2 – Face2Poly conversion

- Slow face to polygon construction times
  - Initial: 4 hours 30 minutes
- Bottlenecks identified through gathering stats:
  - Single sqlplus session approach;
  - Slow insert times;
  - Slow Oracle Spatial Query (SDO\_INSIDE);

## Solution

- Parallel process via multiple sqlplus sessions
  - Spawn multiple sqlplus from command line OR
  - Use Oracle PL/SQL Procedure and DBMS\_JOB
- Avoid conflicts between sessions
  - Using simple row stratification algorithm using MOD
- Improve insertion times
  - Use PL/SQL TABLE OF ROWTYPE and FORALL inserts
- Improve SDO\_INSIDE query times
  - Pin index and table data
  - First\_Rows(2) performance hint (no stats on tables)
- Result – 45 minutes

## Customer 3 – Data Timeliness

- Critical issues:
  - Large data volumes;
    - Predominantly external data;
    - Question marks about actual quality of external data.
  - Unsustainable level of hardware processing and personnel resources being spent on processing external data.
    - Dependence on custom PL/SQL, Materialized Views and SDO\_AGGR\_UNION to deliver products from base (level 0) data.
    - Need to liberate precious staff resources to concentrate on “core business” eg identify non-navigable disconnected roads.
  - Incremental updates
    - Need to maintain vertical topology synchronisation with derived and hierarchical data.
  - Data Auto-Clean because > 0% error rate too high!
    - Response: “Flattening” or “Disaggregation” process.

## Customer 4

- Resource issues limiting attention span of participants;
- Geospatial / IT cultural divide limiting uptake;
  - Knowledge and Tools silos
- Low training investment;
- Number of errors / total features > 0% is too high;
- This customer wants “auto-clean” with rules discovery post-cleaning eg:
  - Find all areas composed of more than one face!
  - Find faces with no associated business feature.
- Lack of “integrated technology stack” in loosely coupled environment → Is data level integration enough?

## Radius Studio

- Goal of database based management is supposed to be about creating fully self-referential data models which do not require applications for their interpretation.
- Currently we do not have spatially-aware self-documenting metadata repositories within the object relational databases within which we store our data.
- Yet even if we did, would we use it?
  - Probably not
- Why?
  - We have had repositories, at the heart of database technologies for in excess of 30 years, but ..
  - Practice and theory are *strangers in the night!* – The doers/thinkers problem
- Model Driven Architecture

## Conclusion

- Why lack of self-referential data models?
  - Conceptual approach (eg normalisation theory, UML, Use Cases....) – not domain-expert friendly
  - Foreign tools eg Data Modelling tools – not “domain” tools (cf customer 4)
  - Lack of education and training? – budgetry issues, course construction difficulties

## Conclusion

- Why Studio?
  - Addresses the “think”/“do” disconnect
  - Useability (cf SQL)
  - Time-invariant “rules” management
    - Software churn
    - Flexibility in implementation
    - Code generation
  - But... it might just enable a fundamental shift in training and software ROI

## Conclusion

- As a Technical Manager I wanted to see my staff being given:
  - Appropriate Tools
  - Good solid education
  - Training investment
- Because... they are the agents of ROI!