

Unit #11

MIS5203

Implementation and Testing

Agenda

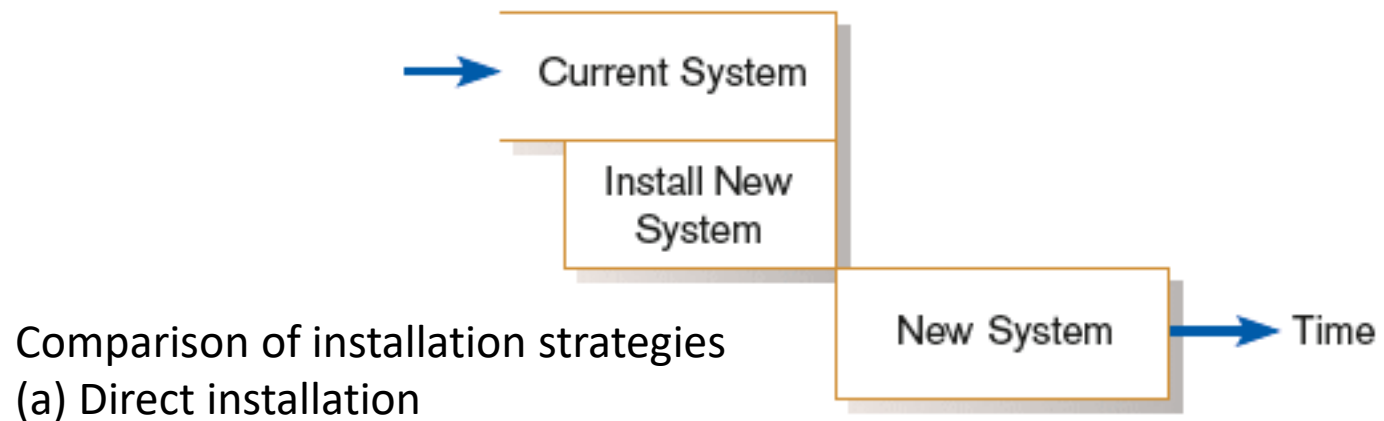
- Installation & change management
- Software applications testing
- Quality characteristics
- Error detection techniques
- Entity inspection
- Evaluating datasets
- Issue tracking

Installation

- **Installation:** the organizational process of changing over from the current information system to a new one
- Four installation strategies:
 - Direct Installation
 - Parallel Installation
 - Single-location installation
 - Phased Installation

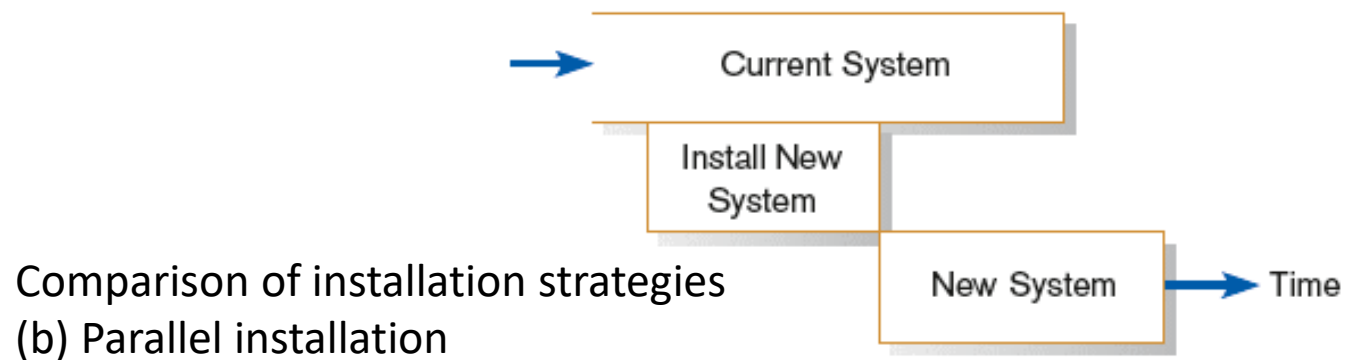
Direct Installation

- **Direct installation:** changing over from the old system to a new one by turning off the old system when the new system is turned on



Parallel Installation

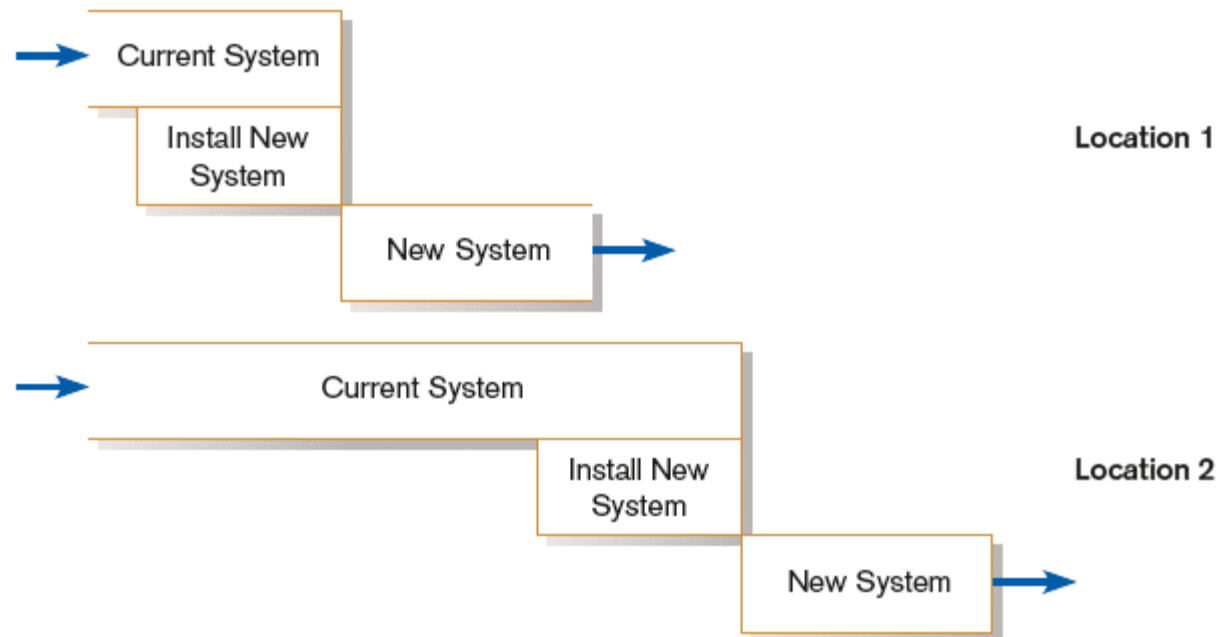
- **Parallel installation:** running the old information system and the new one at the same time until management decides the old system can be turned off



Single-Location Installation

- **Single-location installation:** trying out an information system at one site and using the experience to decide if and how the new system should be deployed throughout the organization
- Also known as location or pilot installation

Single-Location Installation (cont.)



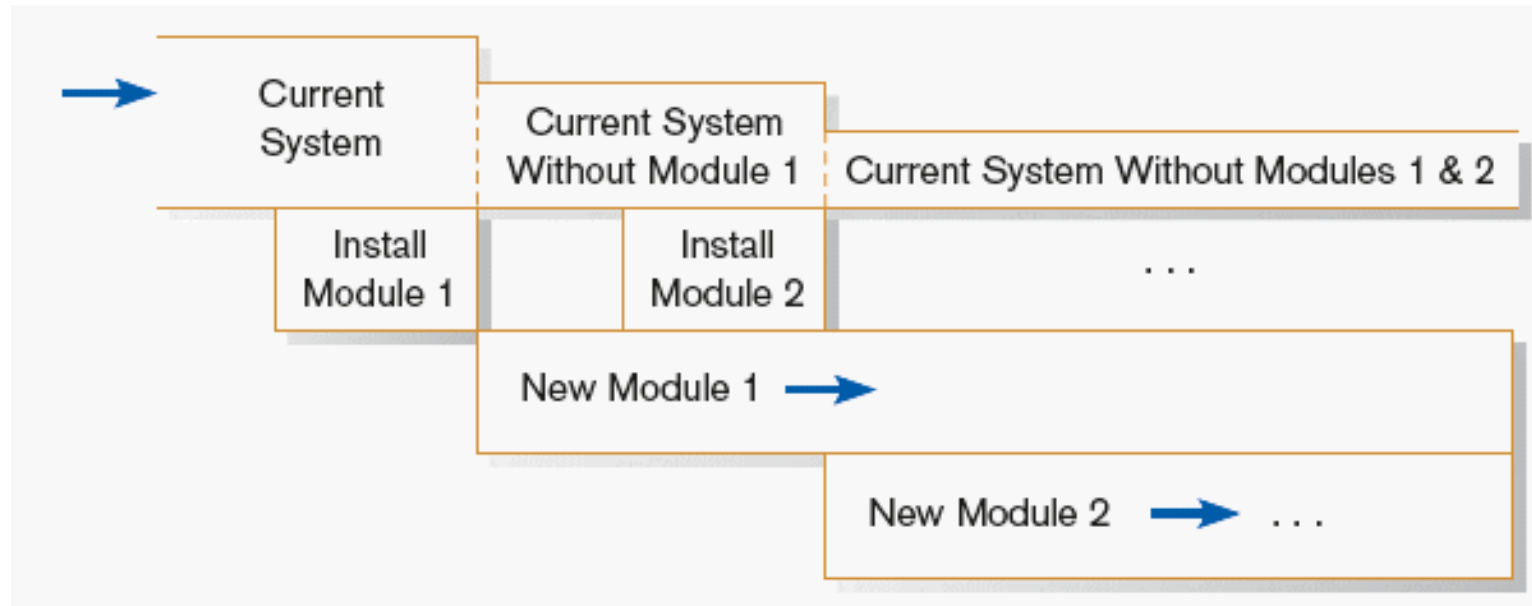
Comparison of installation strategies

c) Single-location installation (with direct installation at each location)

Phased Installation

- **Phased Installation:** changing from the old information system to the new one incrementally, starting with one or a few functional components and then gradually extending the installation to cover the whole new system

Phased Installation (cont.)



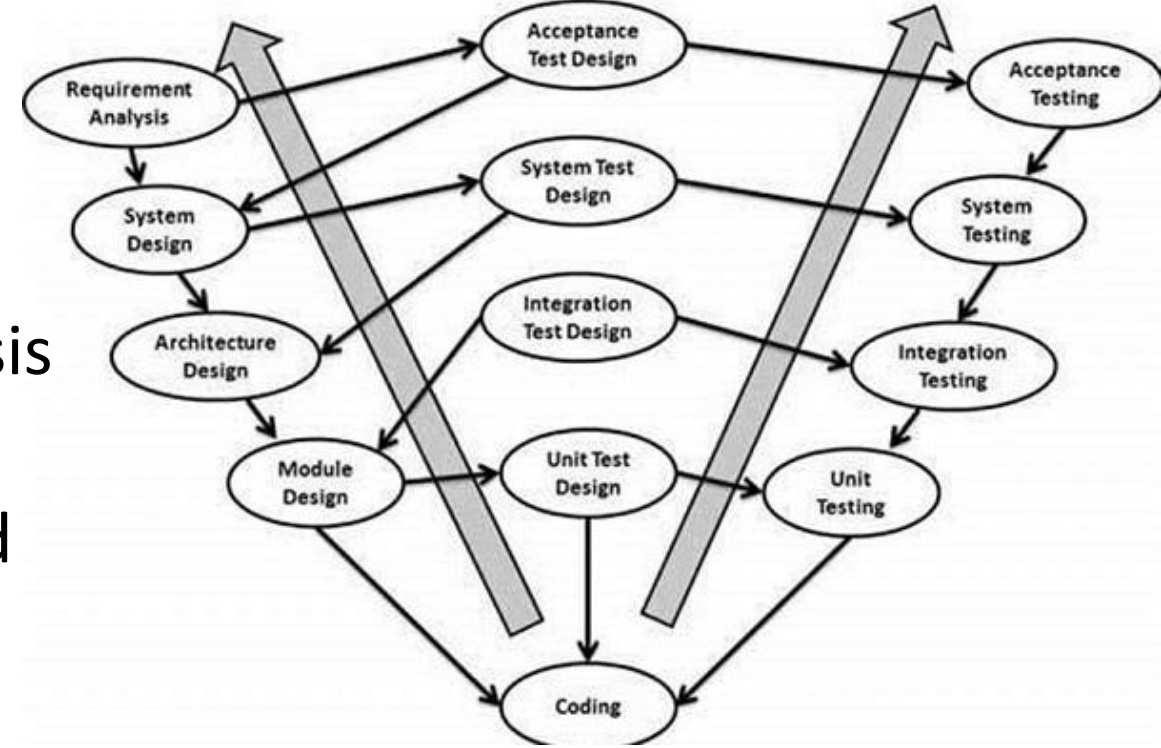
Comparison of installation strategies
(d) Phased installation

Agenda

- ✓ Installation & change management
- Software applications testing
- Quality characteristics
- Error detection techniques
- Entity inspection
- Evaluating datasets
- Issue tracking

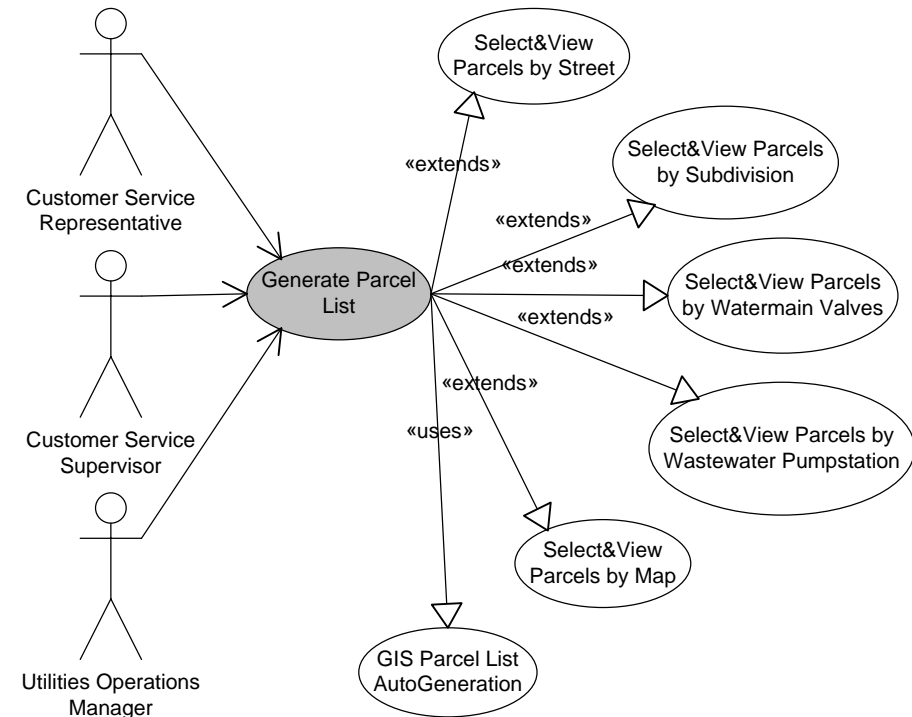
Software Application Testing

- A test plan is developed during the analysis phase
- During the design phase, unit, system and integration test plans are developed
- The actual testing is done during implementation
- Written test plans provide improved communication among all parties involved in testing



Testing Applications

- Testers should be different people than the developers
- Testers should use simulate the various ways end users will use the application, and document errors they find
- Use cases are helpful in developing test suites of alternative test cases



Testing Process

- The purpose of testing is to confirm that the system satisfies the requirements
- Testing must be planned
 - **Bottom-up testing** (most systems are tested early using bottom up testing)
 - Begins testing the smallest units of the system (e.g. programs and modules), and works upward until a the entire system has been tested
 - Advantages: Can be started before all programs are complete; Errors in critical modules can be found early
 - **Top-down testing**
 - Begins testing the breadth and works into the depth of the system
 - Advantages: Tests of major functions and processing conducted early; Interface errors can be detected sooner; Confidence raised in system by seeing a working system

Phases of Testing



- **Pre-Alpha:**- Software is a prototype. UI is complete. But not all features are completed. At this stage, software is not published
- **Alpha:** Software is near its development and is internally tested for bugs/issues
- **Beta:** Software is stable and is released to a limited user base. The goal is to get customer feedback on the product and make changes in software accordingly
- **Release Candidate (RC):** Based on the feedback of Beta Test, you make changes to the software and want to test out the bug fixes. At this stage, you do not want to make radical changes in functionality but just check for bugs. RC is also put out to the public
- **Release:** All works, software is released to the public

<https://www.guru99.com/alpha-beta-testing-demystified.html>

Acceptance Testing



Process where actual users test a completed information system, the end goal is the users' acceptance of it

- **Alpha testing** is carried out in a lab environment and usually, the testers are internal employees of the organization
 - This kind of **testing** is called **alpha** only because it is done early on, near the end of the development of the software, and before **beta testing**
- **Beta Testing** of a product is performed by "real users" of the software application in a "real environment" and can be considered as a form of external User Acceptance Testing
 - Beta version of the software is released to a limited number of end-users of the product to obtain feedback on the product quality
 - Beta testing reduces product failure risks and provides increased quality of the product through customer validation
 - It is the final test before shipping a product to the customers. Direct feedback from customers is a major advantage of Beta Testing. This testing helps to tests the product in customer's environment

Other Types of Testing

- **Pilot testing** – Preliminary test that focuses on specific predetermined aspects of the system. Not intended to replace other testing methods, but to provide a limited evaluation of basic functionalities of the system
- **White box testing** – Assesses the effectiveness of software program logic. Test data used to determine procedural accuracy or conditions of specific program logic (applies to unit and integration testing). Used in a focused manner as exhaustive white-box testing is often cost prohibitive
- **Black box testing** – Integrated testing of the an information system’s functional effectiveness without regard to any specific internal program structure. Applicable to integration (interface) and user acceptance testing
- **Functional/validation testing** – A form of system testing that evaluates functionality against detailed requirements to trace back to customer requirements (did they build the right product)
- **Regression testing** – Testing against use-cases in test plan to assure that changes did not introduce new errors
- **Parallel testing** – Testing the same data within the original system (which will be replaced) and the new systems and comparing the results
- **Sociability testing** – Tests to confirm that he new system can operate in its target environment without adversely impacting existing systems. Focuses on application processing examining the interfaces with other systems that can be running at the enterprise and within the user’s desktop application environment and with the user’s browser

Internal Quality Characteristics

Programmers care about internal characteristics of application quality, as well as external characteristics

1. Maintainability
2. Flexibility
3. Portability
4. Reusability
5. Readability
6. Testability

McConnell, Steve (2004), Code Complete, Microsoft Press

Internal Quality Characteristics

1. **Maintainability:** Ease of revising or fixing an geospatial application
2. **Flexibility:** Ease of extending a geospatial application to support new uses
3. **Portability:** Ease of modifying a geospatial application to operate in a new environment
4. **Reusability:** Ease with which parts of the geospatial application can be reused
5. **Readability:** Understandability of the source code
6. **Testability:** Degree to which a geospatial application can be verified to meet requirements

External Quality Characteristics

Users care that applications are:

- ***Are easy to use, not if they are easy for developers to modify***
- ***Work correctly, not about the structure and readability of the code***

External Quality Characteristics

1. Correctness
2. Usability
3. Efficiency
4. Reliability
5. Integrity
6. Adaptability
7. Accuracy
8. Robustness

External Quality Characteristics

1. **Correctness:** Accuracy of specification, design, data and implementation
2. **Usability:** How easy it is for user to perform their work
3. **Efficiency:** Appropriate use of system resources (memory, execution time, storage, bandwidth)
4. **Reliability:** How well the application performs under prolonged use
5. **Integrity:** Prevents unauthorized or improper access to its programs and its data
6. **Adaptability:** Can be used, without modification, in other systems
7. **Accuracy:** Produces valid results
8. **Robustness:** Continues to function with invalid inputs or a stressful environmental

External quality issues are:

1. Defects, errors, or bugs
2. Omissions
3. Usability problems

...that negatively affect user satisfaction, application acceptance, and project completion!

Error Detection Techniques...

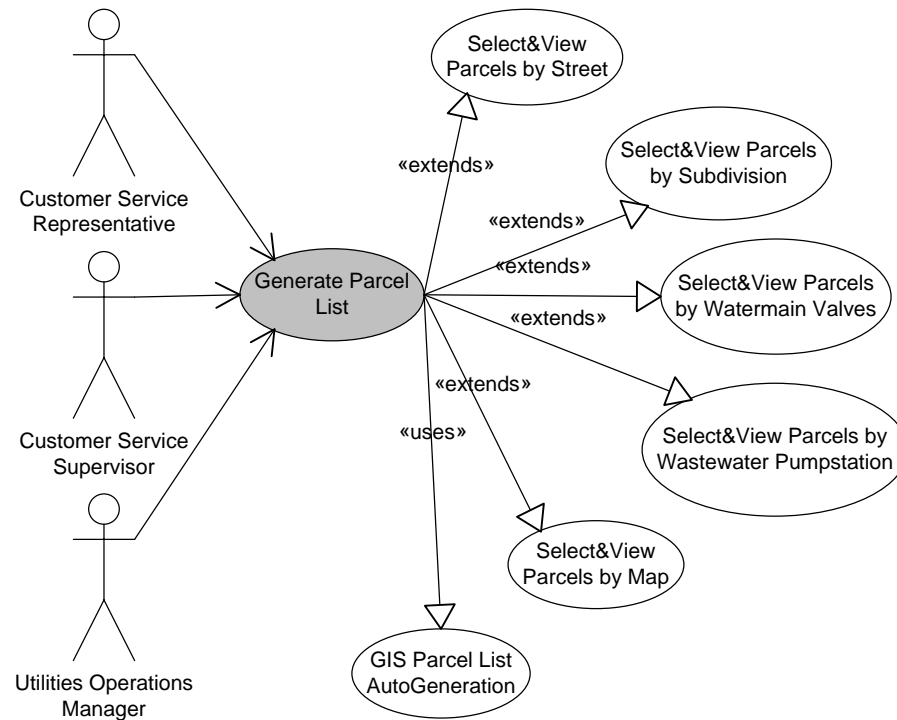
No single defect-detection technique is effective by itself

- *Reviewing designs*
- *Evaluating datasets*
- *Testing the completed applications*

Reviewing designs...

- *Use cases*
- *Data models*
- *Data flow models*
- *Business process models*
- *System architecture models*

Use case specifications

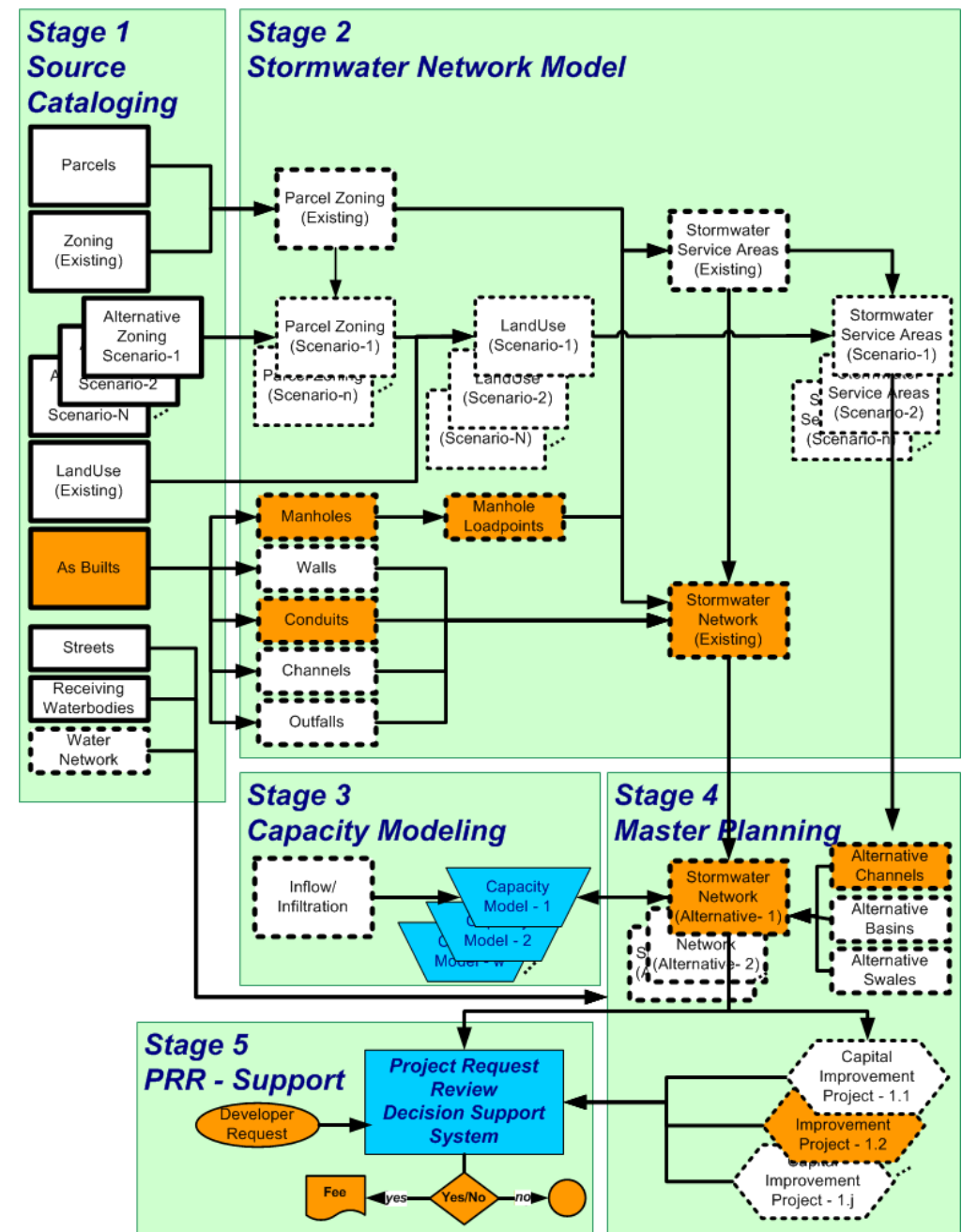


Help identify if the functional specifications are complete, and the desired behavior achieved

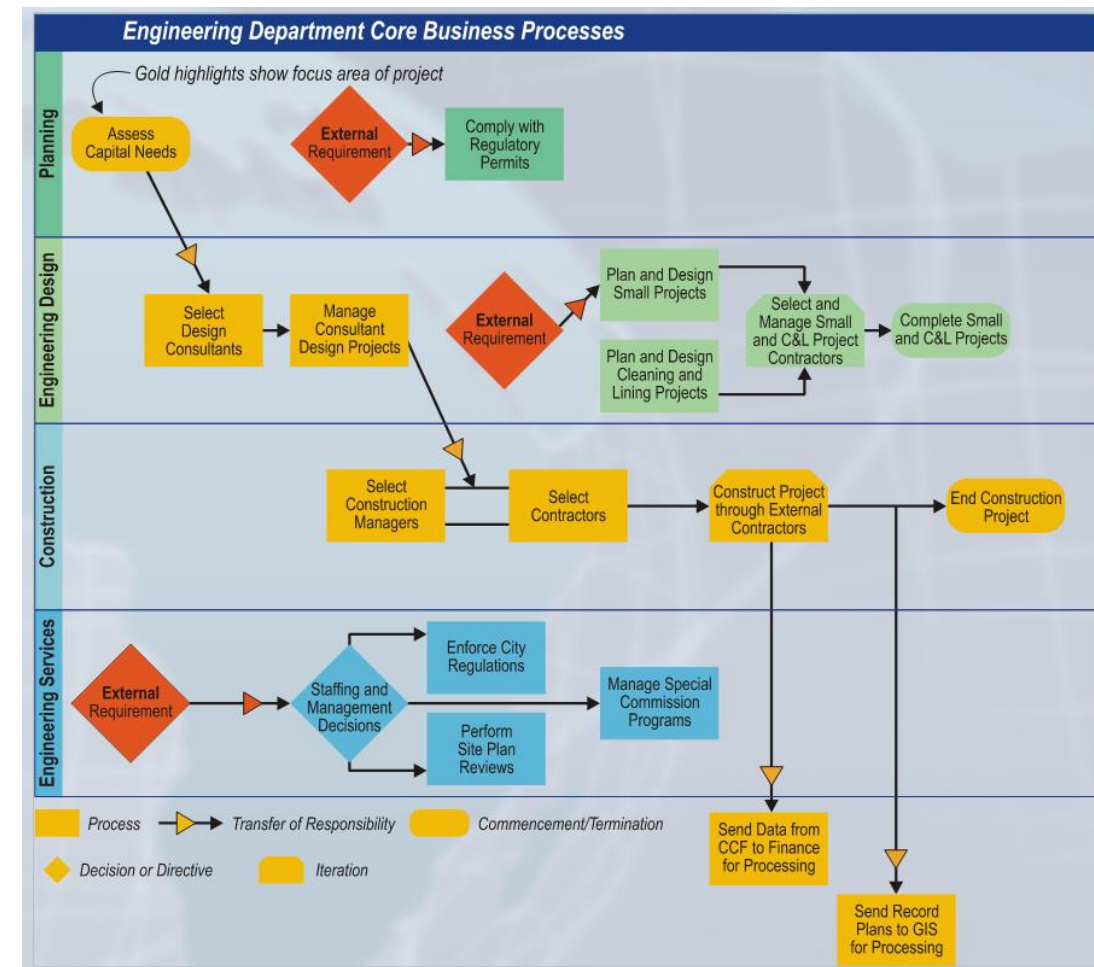
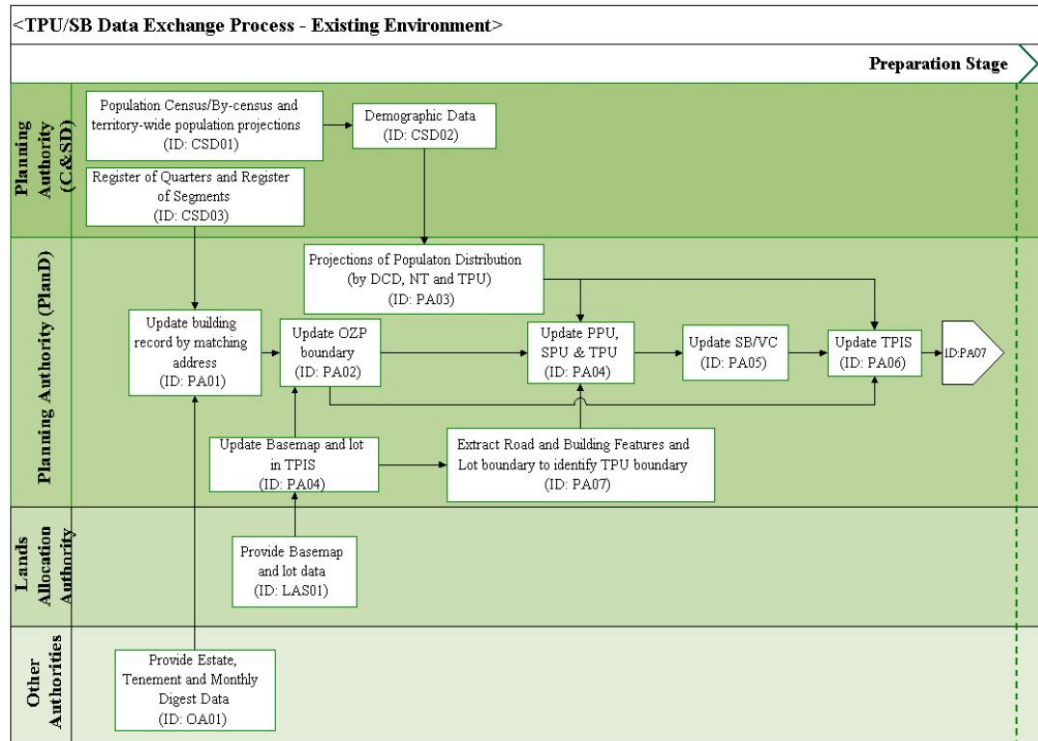
Use Case ID:	3		
Use Case Name:	Generate Parcel List		
Iteration:	Filled		
Created By:	Jennifer Mattie	Last Updated By:	David Lanter
Date Created:	6-26-2005	Date Last Updated:	07-06-2005
Actor:	Customer Service Representative (CSR) Customer Service Supervisor (CSS) Utilities Operations Manager (UOM)		
Description:	The User (CSR, CSS, or UOM) creates a list of parcels affected by an outage affecting one or more sewer system pump stations.		
Triggers:	Notification of an ongoing sewer system pump station outage event or one being planned due to maintenance.		
Preconditions:	<ul style="list-style-type: none"> The pump stations' statuses are up-to-date. GIS Data Server is online. GIS Web Server is online. 		
Postconditions:	A list of parcels for the User to review and submit.		
Priority:	High		
Frequency of Use:	Moderate		
Normal Course of Events:	<ol style="list-style-type: none"> User receives notification of an ongoing or planned sewer system outage event. User invokes the Outage Notification application. User chooses to generate a list of parcels automatically from the GIS (Use Case 10 – GIS Parcel List <u>Autogeneration</u>). 		
Alternative Course:	<ol style="list-style-type: none"> User receives notification of an ongoing or planned sewer system outage event. User invokes the Outage Notification application. User chooses to interactively create a new list of selected parcels, or add/delete parcels to/from the existing selected parcels list by using one or some combination of the Parcel Select and View tools: <ol style="list-style-type: none"> Use Case 4: by Street Use Case 5: by Subdivision Use Case 6: by Water Main Valves Use Case 7: by Wastewater <u>Pumpstation</u> Use Case 8: by Map 		
Exceptions:	None		
Extensions:	Use Cases 4-8		
Includes(Uses):	Use Case 10 – GIS Parcel List <u>Autogeneration</u>		
Related Business Rules:	None		
Special Requirements:	None		
Assumptions:	<ul style="list-style-type: none"> Pump stations' status attribute values stored within the GIS <u>geodatabase</u> are correct and up-to-date. 2. Use Case 10's assumptions concerning GIS data dependencies are met 		
Notes and Issues:	None.		

Data Flow Models

Help identify if data sources and processing logic are valid



Business Process Models



Help identify if the departmental tasks and cross organizational workflows are well understood

Defect Detection Techniques...

- *Reviewing designs*
- **Evaluating datasets**
- *Prototyping*
- *Inspecting software code*
- *Testing applications*

Evaluating Datasets

Three Techniques:

1. *Automated checks*
2. *Visual inspection*
3. *Topology testing*

Automated Checks

- Domain Checks (including Valid Values)
- Nulls and Zeros Checks
- Uniqueness Check
- Default Value Checks
- Duplicate Geometry Check
- Functional Dependency Check

Defect Detection Techniques

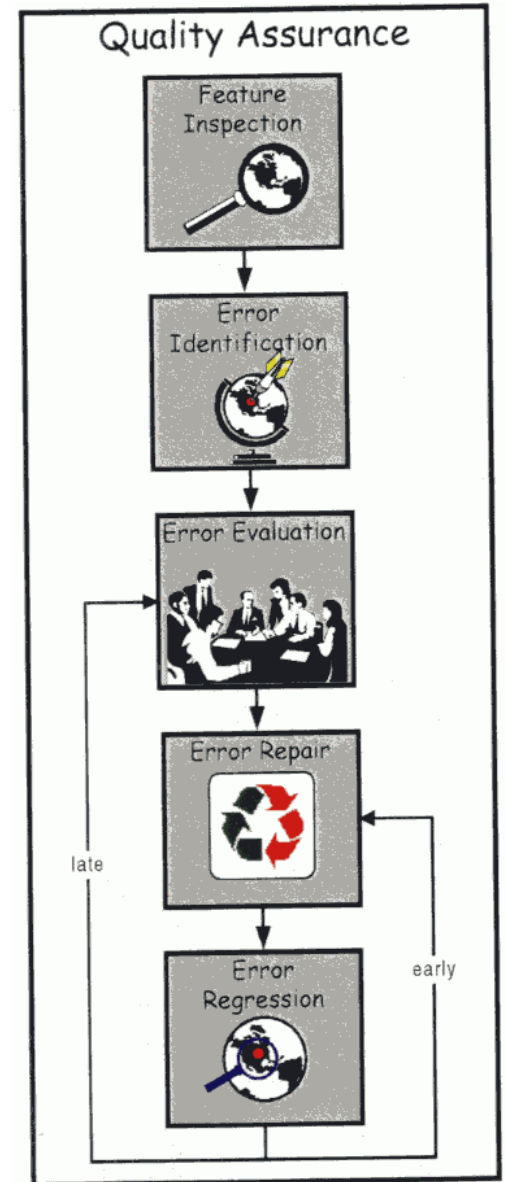
Evaluating Datasets

Three Techniques:

- ✓ *Automated checks*
- 2. Visual inspection**
3. *Topology testing*

Visual Inspection

- *Entities inspected*
 - *each found error is identified and documented*
- *Severity and priority of each error is determined*
 - *cost and risk that repairs will cause further problems are evaluated*
- *Repaired errors are “regressed” to confirm fix*
 - *and to see if fix caused additional errors*



Quantifying Inspection Needs

Entities

	lines	points	nodes	Total
waste	170,000	12,100	167,900	350,000
storm	57,600	28,000	75,350	160,950
water	481,900	173,200	1,489,700	2,144,800
hpfs	8,757	9,500	8,886	27,143
Total	718,257	222,800	1,741,836	2,682,893

X

Attributes

	lines	points	nodes	Total
waste	43	22	25	90
storm	43	22	25	90
water	31	22	21	74
hpfs	31	22	21	74
Total	148	88	92	328

=

67,450,000 attributes

+ *2,682,893 graphic features*

~70,133,000 data elements

Requiring 1,686 person years (@ 3 minutes to compare each to the value in its source document !)

Note: Data volume estimate for water and sewer utility data automated in Philadelphia

Two Methods for Entity Inspection

1. Entity prioritization

- Prioritized entity lists
- Assures more important entities are examined first (before resources expended)

2. Random sampling

- Statistical approach, based on randomly sampled entities
- Yields objective measures of error rates for each entity type
- Provides objective measures of a product's quality

➤ *If features within a dataset have error rates that are too high, errors are fixed and entities are sampled again, process continues until the desired quality is reached*

Random Sampling

1. Random samples of data selected for inspection
2. Errors counted, and compared to threshold of “acceptability”

ANSI Z4.1 Standard is often used to determine:

- Size of random samples
- Error rates for accepting / rejecting a dataset

AMERICAN NATIONAL STANDARD

*Sampling Procedures and Tables for Inspection
by Attributes*

Prepared by
American Society for Quality Standards Committee
For
AMERICAN NATIONAL STANDARDS COMMITTEE
Z-1 ON QUALITY ASSURANCE

Sponsor and Secretariat
AMERICAN SOCIETY FOR QUALITY

Determining Sample Size

In Z1.4, sample size is a function of dataset size

TABLE I—Sample size code letters

(See 9.2 and 9.3)

Lot or batch size			Special inspection levels				General inspection levels		
			S-1	S-2	S-3	S-4	I	II	III
2	to	8	A	A	A	A	A	B	
9	to	15	A	A	A	A	A	C	
16	to	25	A	A	B	B	B	D	
26	to	50	A	B	B	C	C	E	
51	to	90	B	B	C	C	C	F	
91	to	150	B	B	C	D	D	G	
151	to	280	B	C	D	E	E	H	
281	to	500	B	C	D	E	F	J	
501	to	1200	C	C	E	F	G	K	
1201	to	3200	C	D	E	G	H	L	
3201	to	10000	C	D	F	G	J	M	
10001	to	35000	C	D	F	H	K	N	
35001	to	150000	D	E	G	J	L	P	
150001	to	500000	D	E	G	J	M	Q	
500001	and	over	D	E	H	K	N	R	

Determining Sample Size

In Z1.4, sample size is a function of dataset size and inspection level

SINGLE
NORMAL
PLANS

Table II-A—Single sampling plans for normal inspection (Master table)

(See 9.4 and 9.5)

Sample size code letter	Sample size	Acceptable Quality Levels (normal inspection)																									
		0.010	0.015	0.025	0.040	0.065	0.10	0.15	0.25	0.40	0.65	1.0	1.5	2.5	4.0	6.5	10	15	25	40	65	100	150	250	400	650	1000
		Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re
A	2																										
B	3																										
C	5																										
D	8																										
E	13																										
F	20																										
G	32																										
H	50																										
J	80																										
K	125																										
L	200																										
M	315																										
N	500																										
P	800																										
Q	1250																										
R	2000																										

↓ = Use first sampling plan below arrow. If sample size equals, or exceeds, lot or batch size, do 100 percent inspection.
 ↑ = Use first sampling plan above arrow.
 Ac = Acceptance number.
 Re = Rejection number.

Determining Acceptance/Rejection Error Rates

SINGLE
NORMAL
PLANS

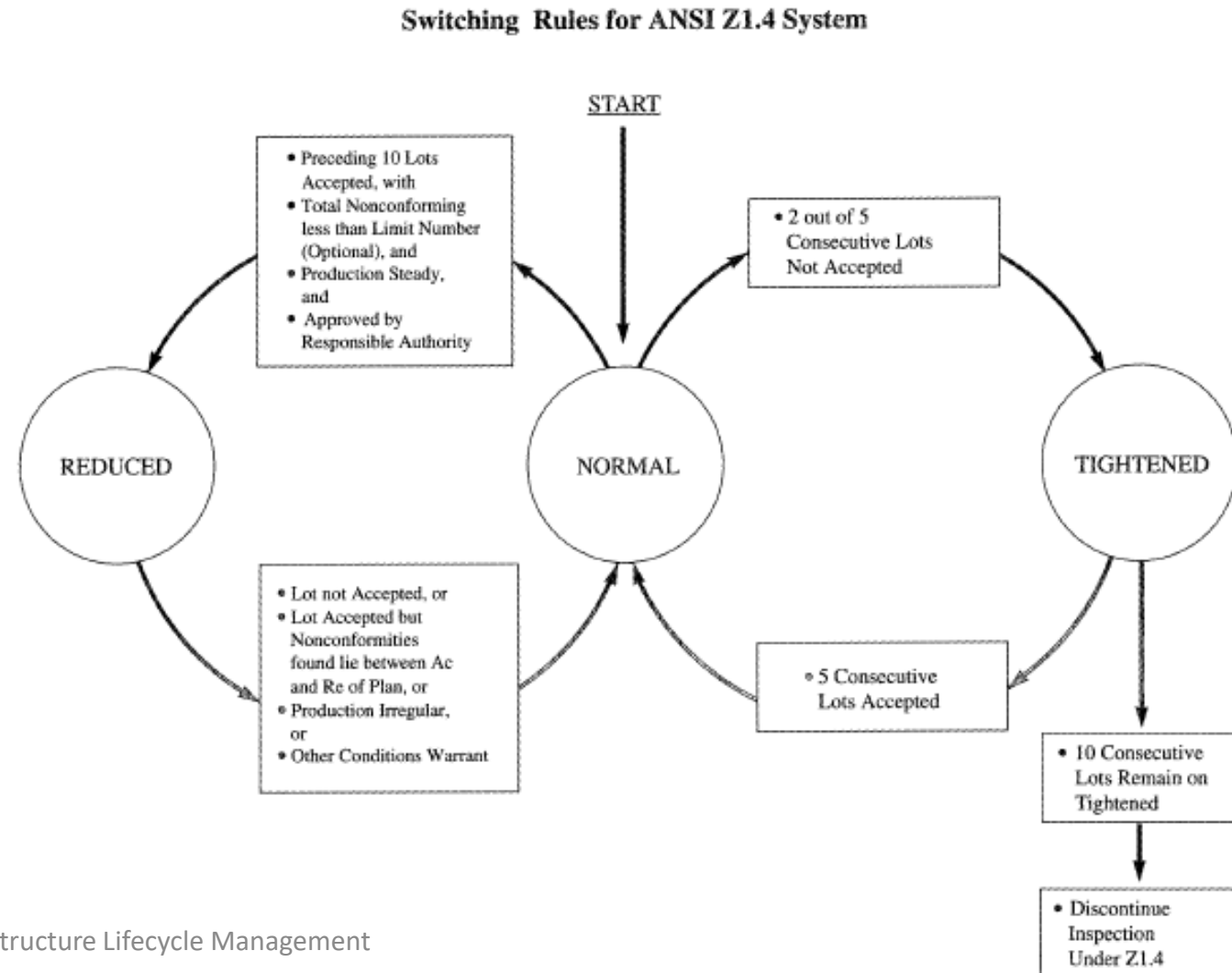
Table II-A—Single sampling plans for normal inspection (Master table)

(See 9.4 and 9.5)

Sample size code letter	Sample size	Acceptable Quality Levels (normal inspection)																									
		0.010	0.015	0.025	0.040	0.065	0.10	0.15	0.25	0.40	0.65	1.0	1.5	2.5	4.0	6.5	10	15	25	40	65	100	150	250	400	650	1000
		Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re	Ac Re
A	2	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
B	3	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
C	5	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
D	8	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
E	15	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
F	20	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
G	52	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
H	80	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
I	125	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
J	200	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
K	315	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
L	500	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
M	800	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
N	1250	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
P	2000	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Q	3150	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
R	5000	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓

↓ = Use first sampling plan below arrow. If sample size equals, or exceeds, lot or batch size, do 100 percent inspection.
 ↑ = Use first sampling plan above arrow.
 Ac = Acceptance number.
 Re = Rejection number.

Rules for Stopping the Inspection and Rejecting a Dataset



Evaluating Datasets

Three Techniques:

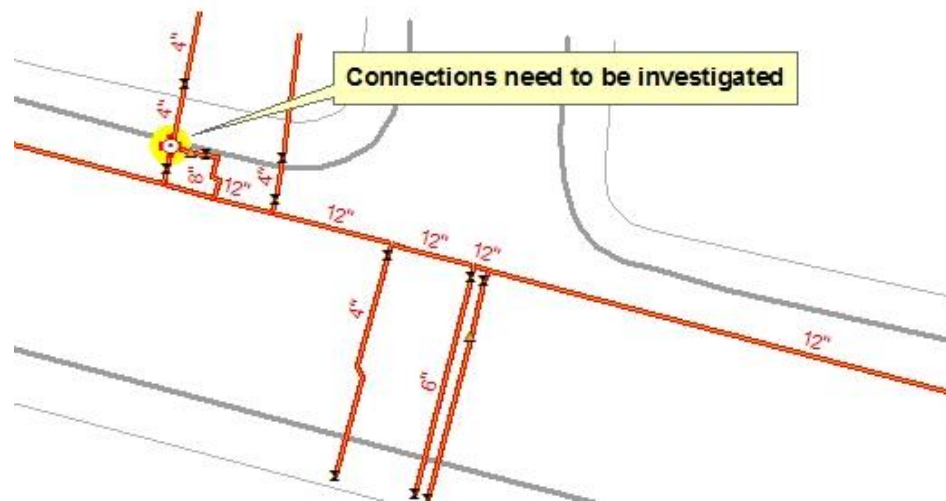
- ✓ *Automated checks*
- ✓ *Visual inspection*
- 3. Topology testing**

Topology Tests

- Cardinality checks
- Valid connections
- Directionality checks
- Gap detection

Topology Tests

Cardinality Testing

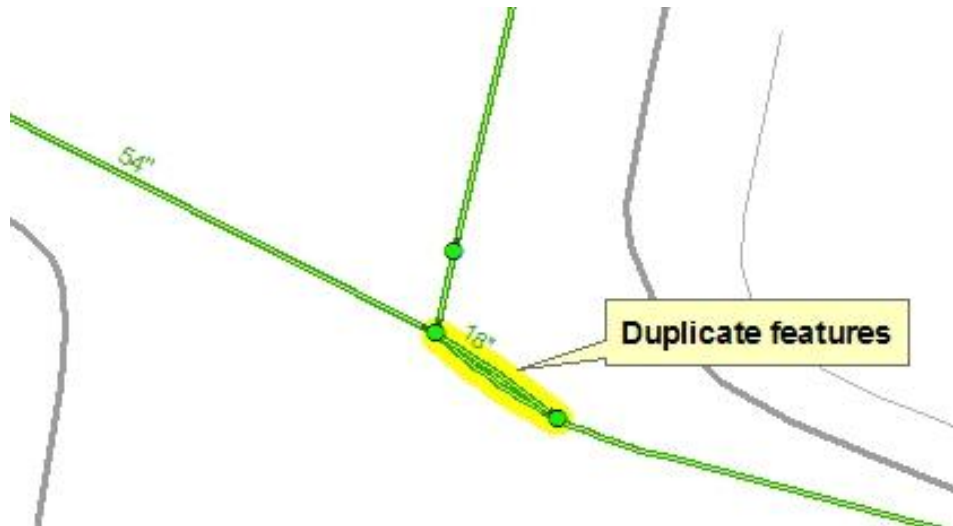


A hydrant is shown with 3 connected pipes.

Typically, a hydrant terminates a single pipe. Source documents will need to be reviewed to see if the hydrant should be a different structure, the pipes need realigned, or if there is a missing hydrant lateral.

Topology Tests

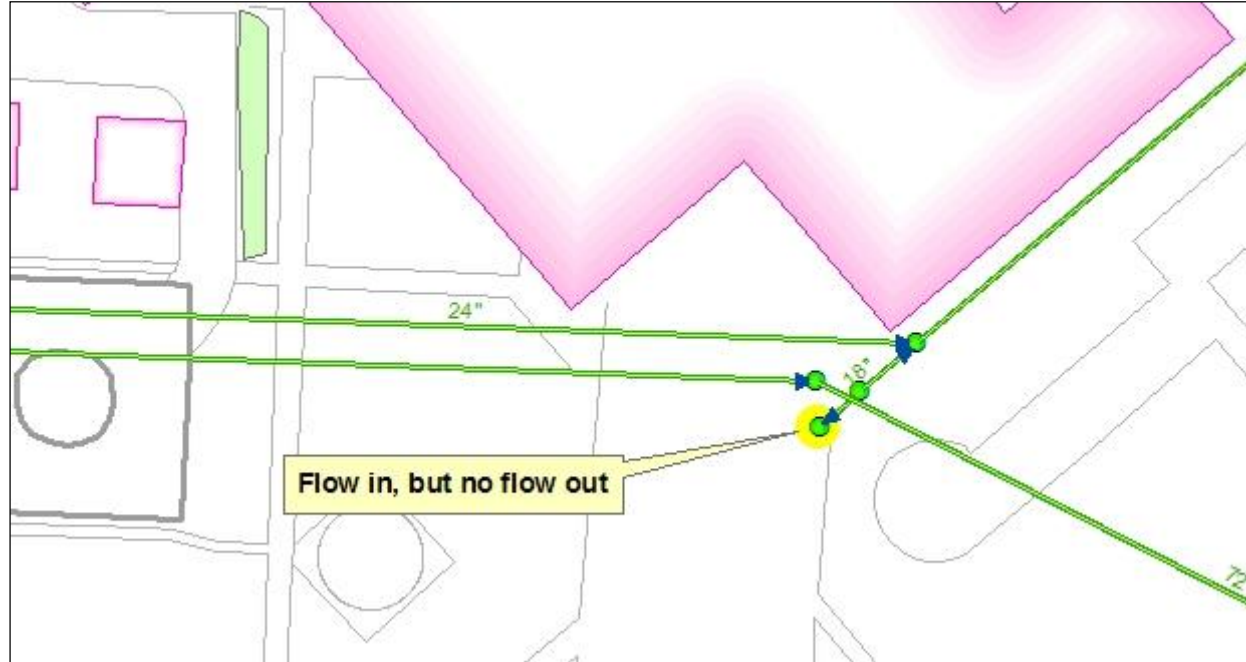
Valid Connections



Duplicate geometry in the sanitary sewer system.

In this case, the From and To nodes share the same coordinates even though the pipes themselves do not share the same digitized geometry

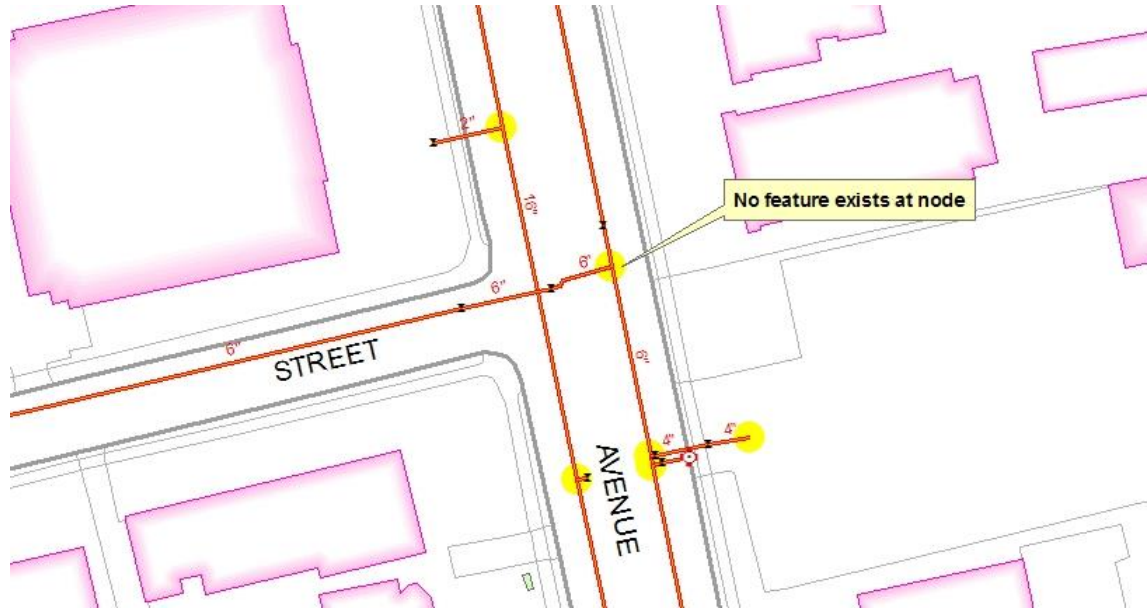
Directionality Testing



A sanitary sewer manhole is shown with a pipe flowing into it, but no pipes flowing out.

Looking at the rest of the system in the area, it is apparent that the connected pipe needs to be flipped to correct the problem..

Gap Detection



Intersections are highlighted where a connection node feature should exist

Most shown here will become tees, but a possible hydrant, cap, or plug may be created on the lower-right node. Source documents may need to be consulted in this case.

Error types and tests to find them:

		Automated Checks	Context Inspection	Random Visual Inspection	Connectivity Testing	Flow Trace Testing
Dataset	Format/Domain Error	●		x		
Source	Missing/Unusable			●		
Feature	FeatureMissing		●	x		
Feature	FeatureShouldNotExist		●	x		
Attribute	AttributeMissing	x		x		
Attribute	AttributeShouldNotExist	x		x		
Attribute	AttributeValueError	x		●		
Attribute	AttributeSourceError			●		
Graphic	LocationError		x	●		
Graphic	ShapeError		x	●		
Graphic	GraphicSourceError			●		
Topology	ConnectionBreak		x	x	x	●
Topology	ConnectionInvalid		x	x	●	x
Topology	DirectionIncorrect			x		●
Annotation	Missing	x	x	●		
Annotation	ShouldNotExist		x	●		
Annotation	LabelWrong		x	●		
Annotation	LocationWrong		x	●		
Annotation	FontWrong		x	●		

Defect Detection Techniques...

- ✓ *Reviewing designs*
- ✓ *Evaluating datasets*
- ✓ *Prototyping*
- ✓ *Inspecting software code*
- **Testing applications**

Issues are:

1. Defects, errors, or bugs
2. Omissions
3. Usability problems

...that negatively affect user satisfaction, geospatial application acceptance, and project completion!

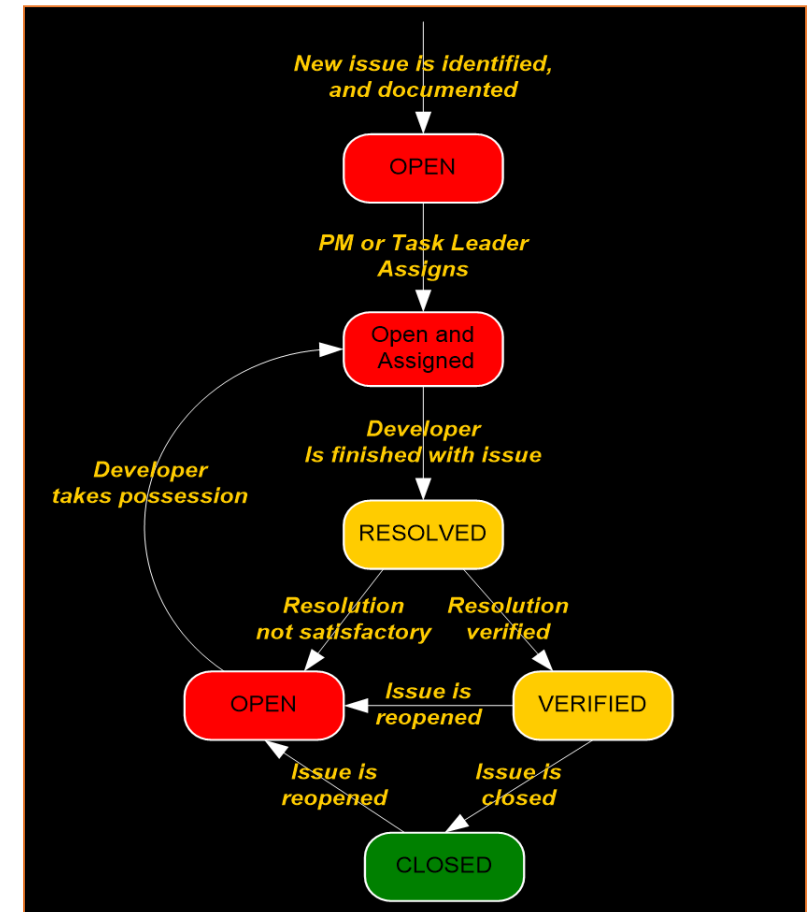
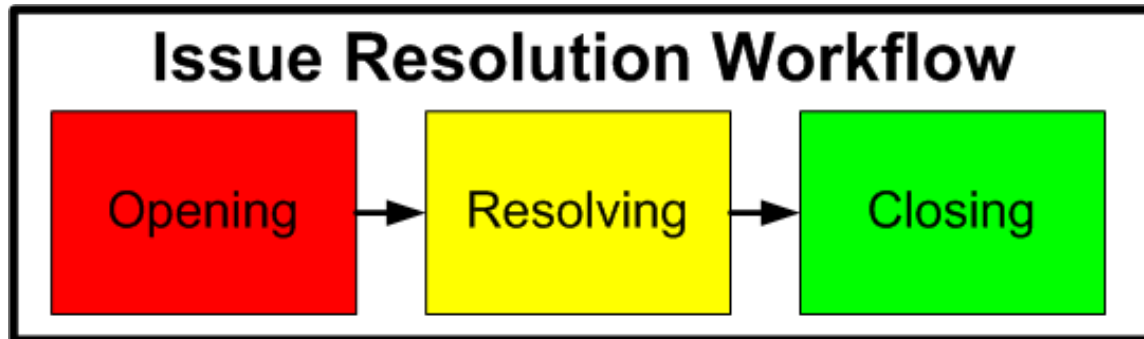
Standard Operating Procedures

- **Issue Resolution Workflow**
- Triage
- Lessons Learned

Issue Resolution Workflow

There are 3 phases in the issue resolution workflow:

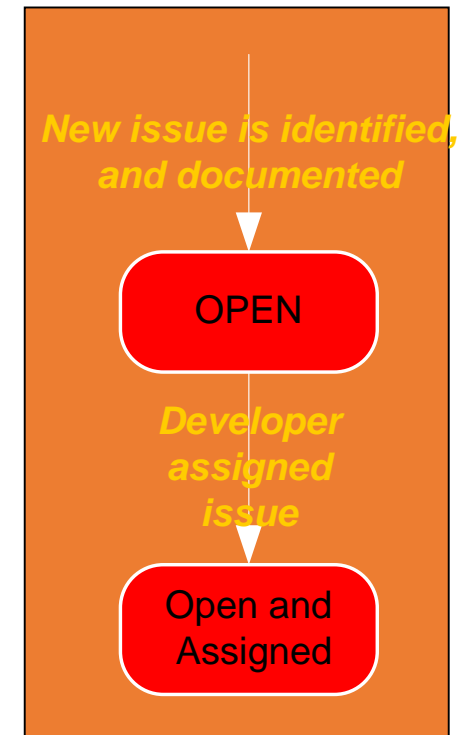
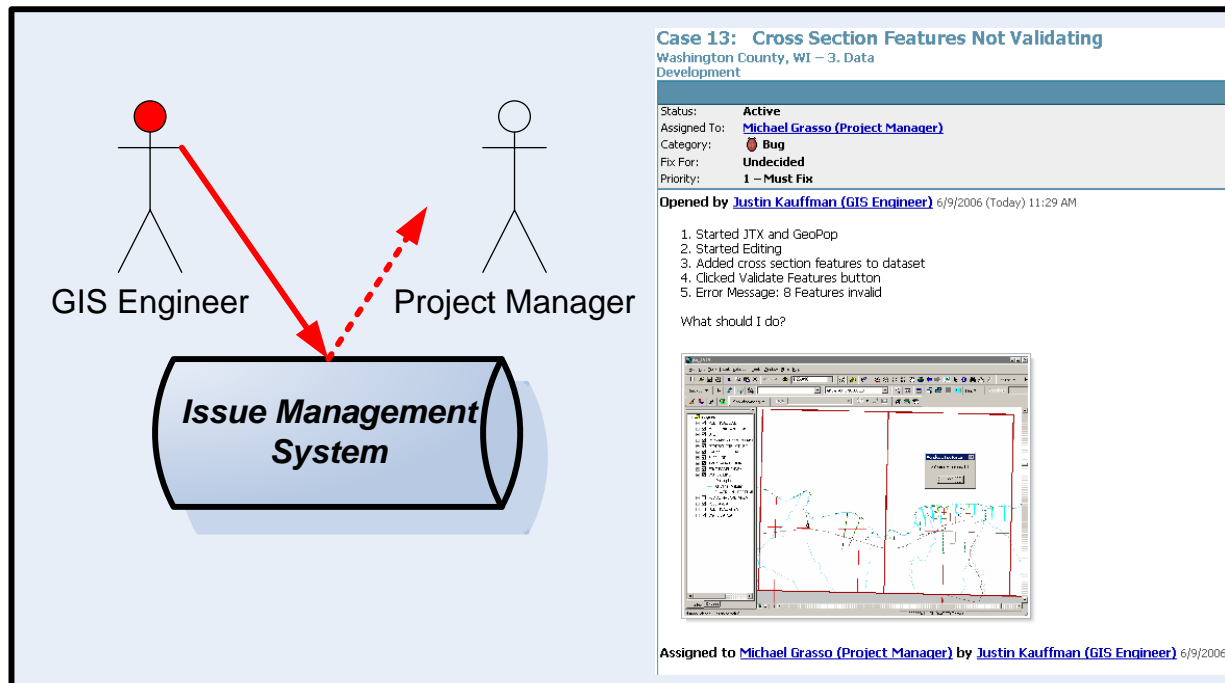
1. Opening
2. Resolving
3. Closing



“Opening” a New Issue

When a new issue is found:

- Create a new issue record form to fill in
- Document the problem
- Save the issue record documenting the problem
- Assign the issue to the person responsible for resolving the issue



Opening a New Issue

- Issues are found by project team members and by end-users
- New issues are opened

For example by clicking on the

[Add New Issue](#) button in the “Issue Manager” system, which responds with the Add/Edit Issue screen, illustrated below:

The screenshot shows a web browser window titled "UltraApps Issue Manager - Microsoft Internet Explorer". The address bar shows the URL "http://oned006002/IssueTracker/IssueNew.asp". The page has a navigation menu with links: "Issue List", "Add New Issue", "My Profile", "Login / Logout", and "Administration". The main content area is titled "Add/Edit Issue" and contains a form with the following fields:

Project	FDOT
Release	R1
Issue Type	
Entity	Admin
Issue Name	
Issue Desc	
How Found	PM
Severity	Sev-1
Priority	Normal
Status	Open
Resolution	
Assigned To	David Brown
Submitted by	David Lanier
Date Submitted	4/23/2004 2:06:23 PM

A "Submit" button is located at the bottom right of the form.

Opening and new issue

FogBugz - Microsoft Internet Explorer provided by Provided by CDM

File Edit View Favorites Tools Help

Address <http://cdm.si-dios.info/FogBugz/default.asp?pre=preSubmitBug>

User: David Lanter

[List](#) [New Case](#) [Send Email](#) [Options](#) [Filters](#) [Discuss](#) [Snippets](#) [Log Off](#)

Open cases by project

ArcIMS Data Viewer	2	0	0
Problem Resolution System	0	0	1

Open cases by client

Open cases by dept

CH2M Hill / CDM Team	0	0	1
--------------------------------------	---	---	---

Open cases by release

Global:

Undecided	2	0	1
---------------------------	---	---	---


Open cases by priority

3-Severity 2 - Must Fix	1	0	0
4-Severity 2 - Must Fix	1	0	1

Open cases by person

David Lanter	0	0	1
Rami Raad	2	0	0

Picture of the day



WELCOME TO **FogBugz**.

[Bug Tracking](#), [Customer Relationship Management](#), and [Message Board](#)

TASKS

[List](#)

[Enter a New Case](#)

[Send Email](#)

[Capture Screenshots](#)

[Search](#)

[Discussion Groups](#)

[Release Notes](#)

Configure [Options](#) and [Snippets](#)

ASSISTANCE

[FogBugz Help](#)

[Contact the Local FogBugz Administrator](#)

You are logged on as **David Lanter**
[\(Log Off\)](#)

SHOW ME:

[Cases](#)

[Page Filters](#)

FogBugz Version 4.0.33 (DB 469)
© Copyright 2000-2005 [Fog Creek Software](#), Inc. All Rights Reserved.

Issue form

The screenshot shows a web browser window titled "FogBugz - Microsoft Internet Explorer provided by Provided by CDM". The address bar contains the URL: `http://cdm.si-dios.info/FogBugz/default.asp?command=new&pg=pgEditBug`. The browser's toolbar includes buttons for Back, Forward, Stop, Home, Search, Favorites, and other navigation functions. The FogBugz interface is displayed, showing the user "David Lanter" and navigation links like "List", "New Case", "Send Email", "Options", "Filters", "Discuss", "Snippets", and "Log Off".

The main form area is titled "Title:" and includes the following fields and options:

- Title:** A text input field.
- Project:** A dropdown menu set to "ArcIMS Data Viewer".
- Area:** A dropdown menu set to "Misc".
- Status:** A dropdown menu set to "Active".
- Assigned To:** A dropdown menu set to "Primary Contact (Rami Raad)".
- Category:** A dropdown menu set to "Bug".
- Fix For:** A dropdown menu set to "Undecided".
- Priority:** A dropdown menu set to "1 - Severity 1 - Must Fix".
- Due:** A date and time selector showing "mm/dd/yyyy" and "hh:mm AM".
- Estimate:** A text input field.
- Version:** A text input field.

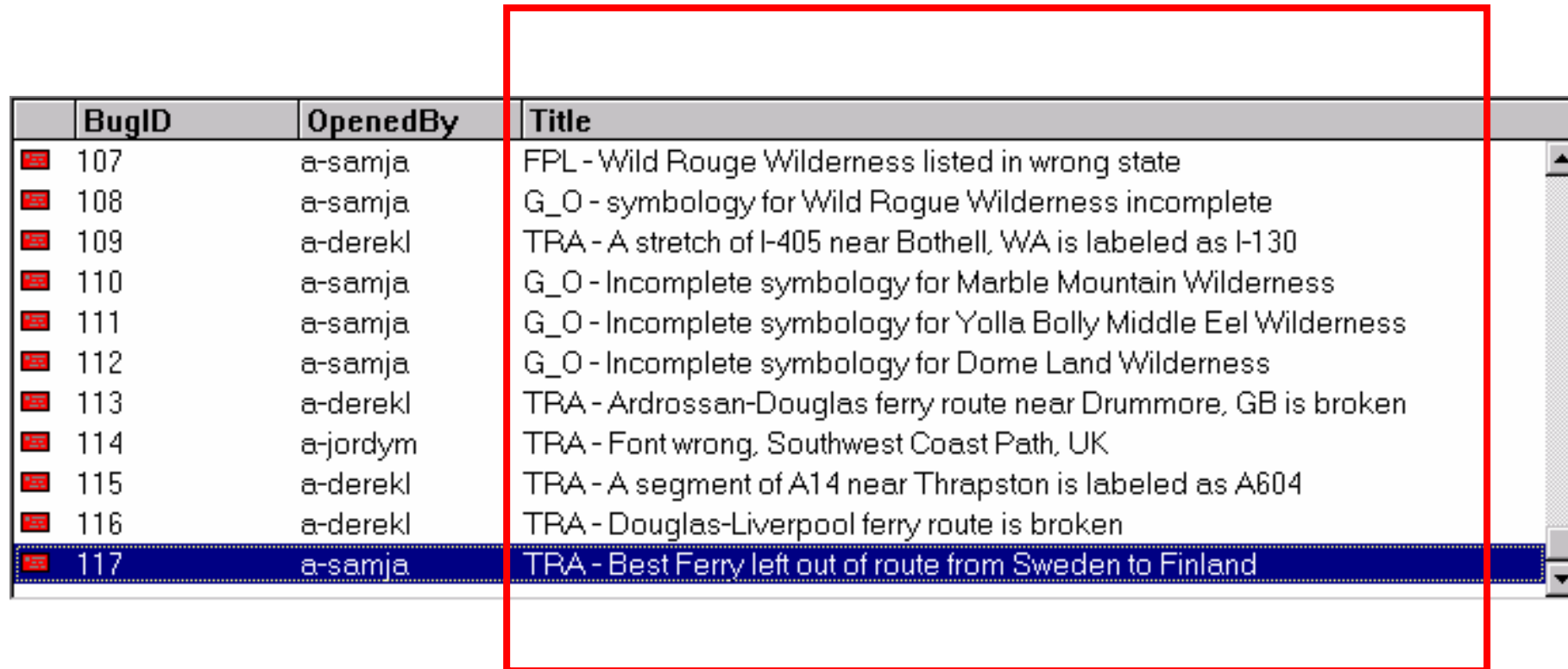
Below the form fields, it indicates the issue was "Opened by David Lanter" on "3/29/2006 (Today) 8:13 PM". There is a large text area for the description, currently containing a yellow box with the text "Title" and "Enter a one line description of the problem." Below this is an "Attach a file:" section with a "Browse..." button. At the bottom of the form are "OK" and "Cancel" buttons. A link for "Capture Screenshots" is also visible.

Information for Opening a New Issue

1. Status
2. Application Name and Version
3. Title
4. Category
5. Area
6. Error Type
7. Severity
8. Notes
9. Assigned To
10. Release
11. Priority
12. Opened By
13. Opened Date/Time
14. IssueID

Title

Opening a New Issue



A screenshot of a bug tracking interface showing a table of issues. The table has four columns: BugID, OpenedBy, and Title. The last row is highlighted in blue. A red rectangular box is drawn around the table, specifically highlighting the Title column. The table contains the following data:

BugID	OpenedBy	Title
107	a-samja	FPL - Wild Rouge Wilderness listed in wrong state
108	a-samja	G_O - symbology for Wild Rogue Wilderness incomplete
109	a-derekl	TRA - A stretch of I-405 near Bothell, WA is labeled as I-130
110	a-samja	G_O - Incomplete symbology for Marble Mountain Wilderness
111	a-samja	G_O - Incomplete symbology for Yolla Bolly Middle Eel Wilderness
112	a-samja	G_O - Incomplete symbology for Dome Land Wilderness
113	a-derekl	TRA - Ardrossan-Douglas ferry route near Drummore, GB is broken
114	a-jordym	TRA - Font wrong, Southwest Coast Path, UK
115	a-derekl	TRA - A segment of A14 near Thrapston is labeled as A604
116	a-derekl	TRA - Douglas-Liverpool ferry route is broken
117	a-samja	TRA - Best Ferry left out of route from Sweden to Finland

Area (continued)

Opening a New Issue

- For example, a Computerized Maintenance Management Systems, might have Areas setup for:
 - User login accounts
 - Authentication and access permissions
 - Service requests
 - Work orders
 - Database
 - Map controls
 - Map data themes, e.g. Assets, BaseMap, Roads, Populated Places, etc.
 - Reports
 - Documentation

Severity

Opening a New Issue

Identifies the importance of the quality issue, four level ranking:

- “Severity 1”
- “Severity 2”
- “Severity 3”
- “Severity 4”

Severity 1

Opening a New Issue

- A “Brand Damaging error”
- Can be a defect which renders the application unusable - a system crash, an unhandled application error, inability to start/stop, failure to login, etc.
- *Do not deliver a product that contains Severity 1 defects*

Severity 2

Opening a New Issue

- “Product Damaging error”
- May be a systematic error, such as a Query subsystem that returns incorrect result sets;
- *Do not a product that contains Severity 2 defects*

Severity 3

Opening a New Issue

- A “Non-show Stopper”
- A simple (non-systematic) error that will be fixed if there is time and resources
- Associated with a work around
- Or, misspelled text message, missing lable, or consistency problem
- *Fix if there is time and budget*

Severity 4

Opening a New Issue

- A “recommendation”
- Is not a defect nor product acceptance issue
- Usually a recommendation or request that falls outside the scope of the current project
- Associated with a work around
- Or, misspelled text message, missing label, or consistency problem
- *Do not fix*

Notes

Opening a New Issue

- The description of the problem
- Extremely important, this is where the person identifies the problem and reports how to reproduce it.
- Every good bug report needs exactly three things in the Notes or Description field:
 - Steps to reproduce
 - What you expected to see
 - What you saw instead (see description below)

Also referred to as “Description” in some issue tracking systems.

Notes

Opening a New Issue

To reproduce:

1. Logged in as: mt491jm
2. Selected: Application/Service/Search/Service Requests
3. Searched by:
 - Depth Of Search = SR
 - Check "Service Request" box
 - Status = Open
 - Priority = High

Click "Go", got a list of SRs on the left panel

Click the link of WO-1329-1-2, the Work Order detail shows up

Click the link of SR-1329 - Got error message: "No matching items were found in the list," Expected to see SR's details

Opened by [Fei Wang](#) 3/29/2006 (Yesterday) 3:18 PM

Reproduce steps:

- 1) go to <http://imserver/wilmington>
- 2) zoom in the map so that you can see buildings and pipes
- 3) click on "redline tool"
- 4) digitize a polygon
- 5) click on Save button on Redling frame
- 6) type in Title, Description, Name, then click on Save Session
- 7) an error message shows up: "Error: adding session information - details: Operation must use an updateable query"

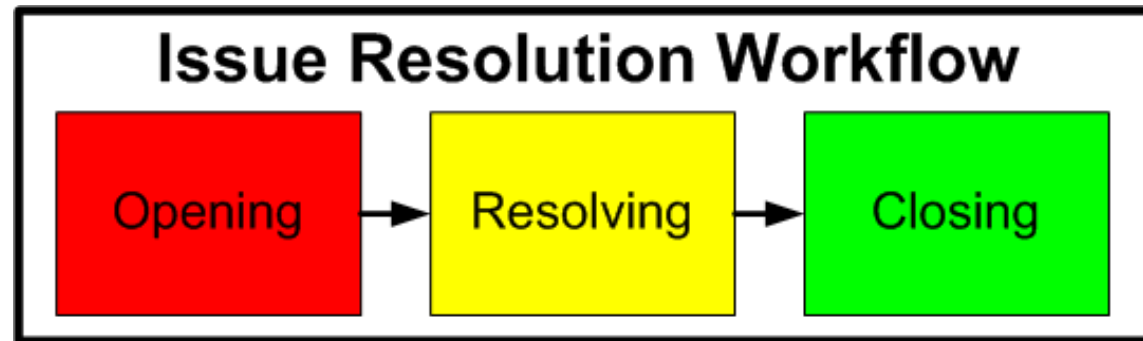
what is an updateable query?

I didn't see any definition in their User Guide document

Issue Resolution Workflow

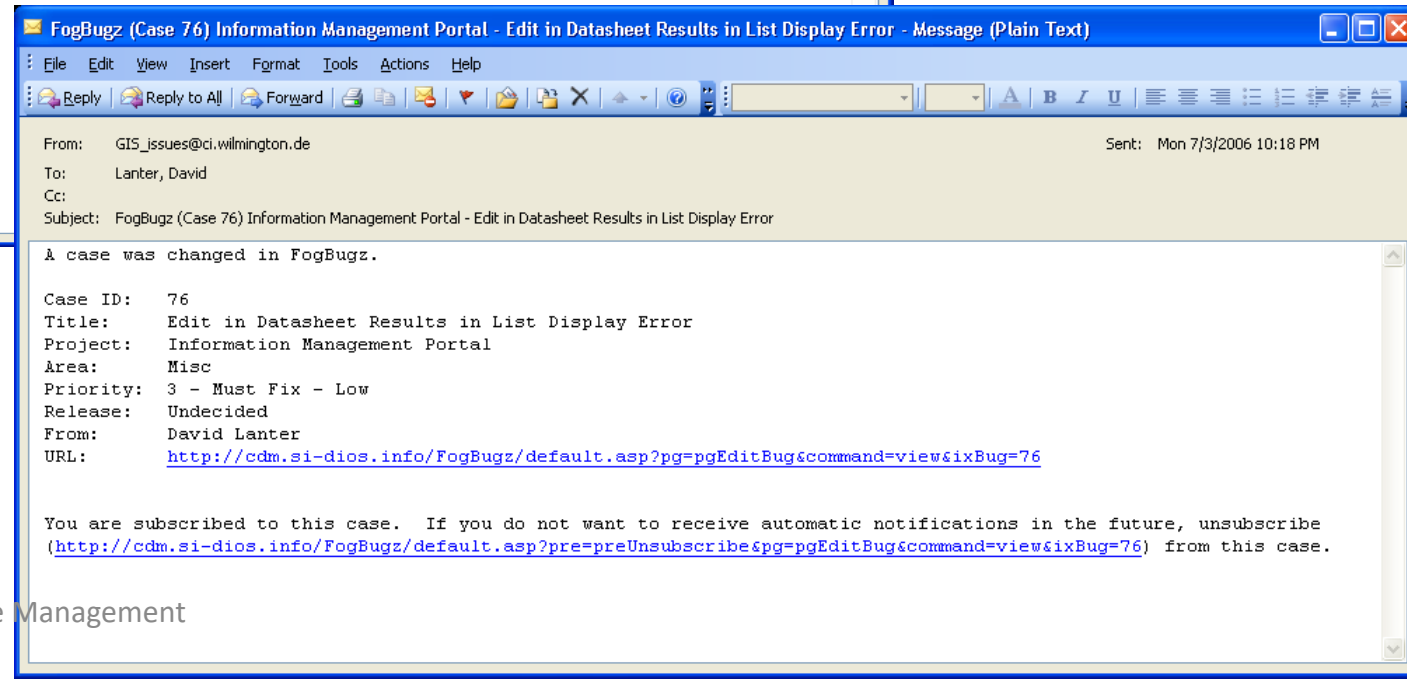
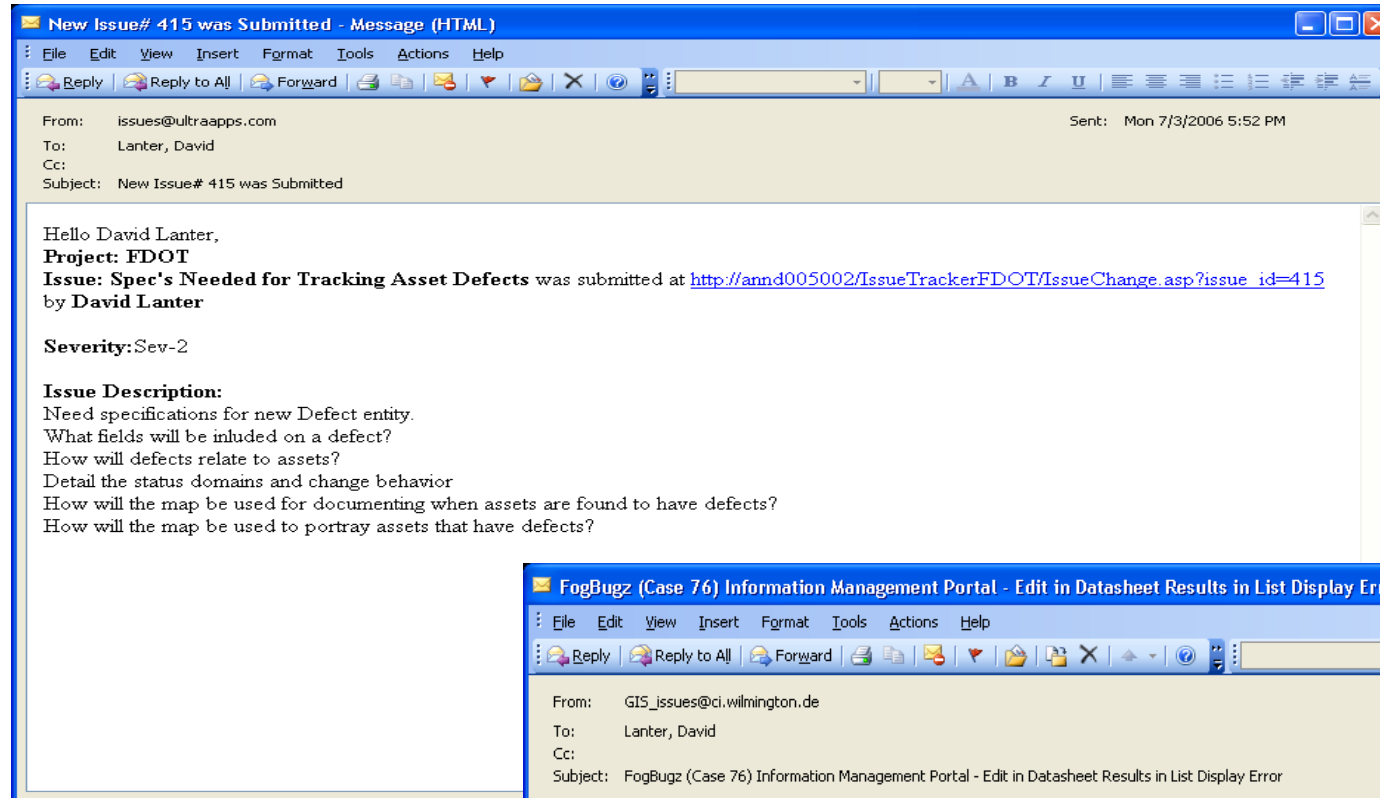
There are 3 phases in the issue resolution workflow:

1. Opening
2. **Resolving**
3. Closing

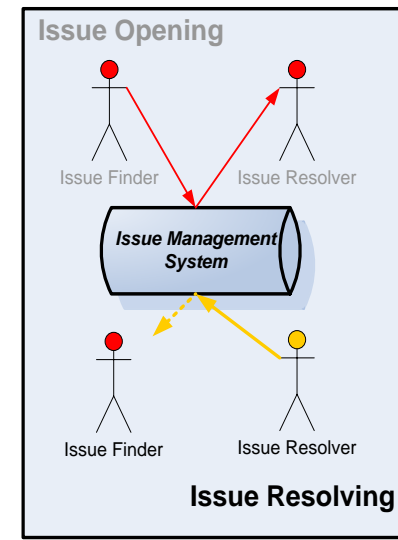


Email Notifications

Opening a New Issue



Resolving an issue

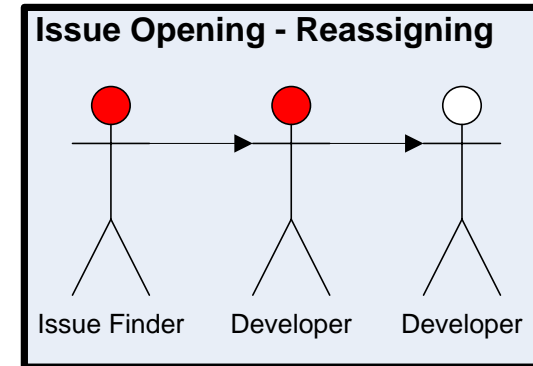


- After you are notified that you have a new issue
- Follow the directions in the notes and attempt to reproduce the issue
- Decide:
 - If the Notes provided enough information to reproduce the issue
 - If the issue is a valid issue
 - If you are the appropriate person to resolve the issue
 - Estimate how long it will take to resolve the issue
- Resolve it in order, by: Release, Severity, and Priority
- Reassign the issue for validation of your resolution

Re-Assigning Issues

- Sometimes, you are assigned issues that should be resolved by someone else
- When this happens, reassign it by changing the “Assign To” value:

Resolving an issue



The diagram shows four stick figures representing roles: GIS Engineer, Project Manager, Program Manager, and FEMA Engineer. The first three have red heads, while the fourth has a white head. Red arrows show the flow: from GIS Engineer to Project Manager, from Project Manager to Program Manager, and from Program Manager to FEMA Engineer. A dashed red arrow also points from the Program Manager to the FEMA Engineer. Below the figures is a blue cylinder labeled 'CDM's Issue Management System'.

Case 14: Required Community ID is Unknown
Washington County, WI – 3. Data Development

Status: **Active**
Assigned To: [Ralph Ellison \(FEMA Engineer\)](#)
Category: **Bug**
Fix For: **Data Complete: 6/16/2006**
Priority: **3 – Must Fix**

Opened by [Justin Kauffman \(GIS Engineer\)](#) 6/9/2006 (Today) 3:27 PM

The Community ID is a required field, but I can not find data for it. ArcMap/GeoPop will not save the feature without it.

What should I do?

Assigned to [Michael Grasso \(Project Manager\)](#) by [Justin Kauffman \(GIS Engineer\)](#) 6/9/2006 (Today) 3:27 PM

Assigned to [Scott Stone \(Program Manager\)](#) by [Michael Grasso \(Project Manager\)](#) 6/9/2006 (Today) 3:28 PM
Fix For changed from 'Undecided' to 'Data Complete: 6/16/2006'.
Priority changed from '1 - Must Fix' to '3 - Must Fix'.

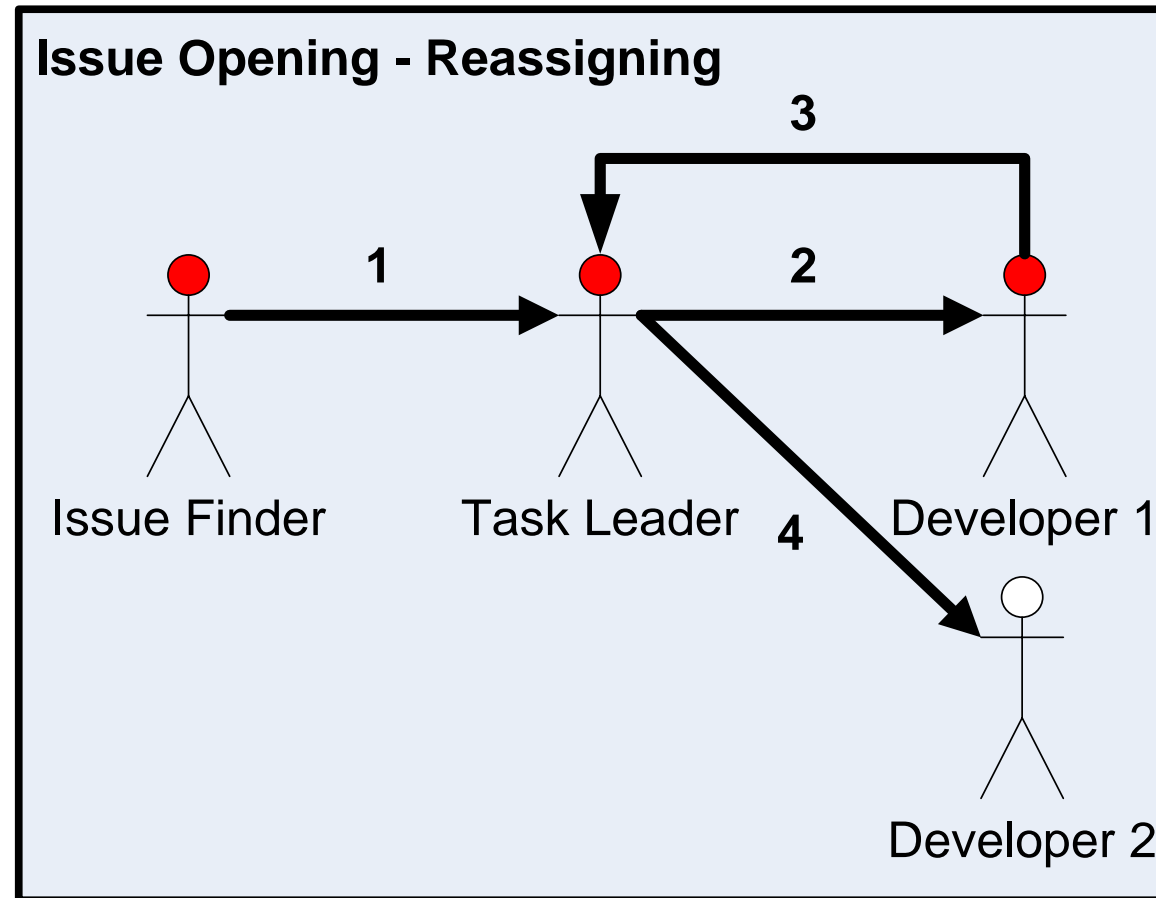
Can we develop the Preliminary datasets without Community ID's on the Admins?

Assigned to [Ralph Ellison \(FEMA Engineer\)](#) by [Scott Stone \(Program Manager\)](#) 6/9/2006 (Today) 3:32 PM
What is FEMA's guidance for handling unknown Community ID's prior to Preliminary Map Production? Can we leave them blank?

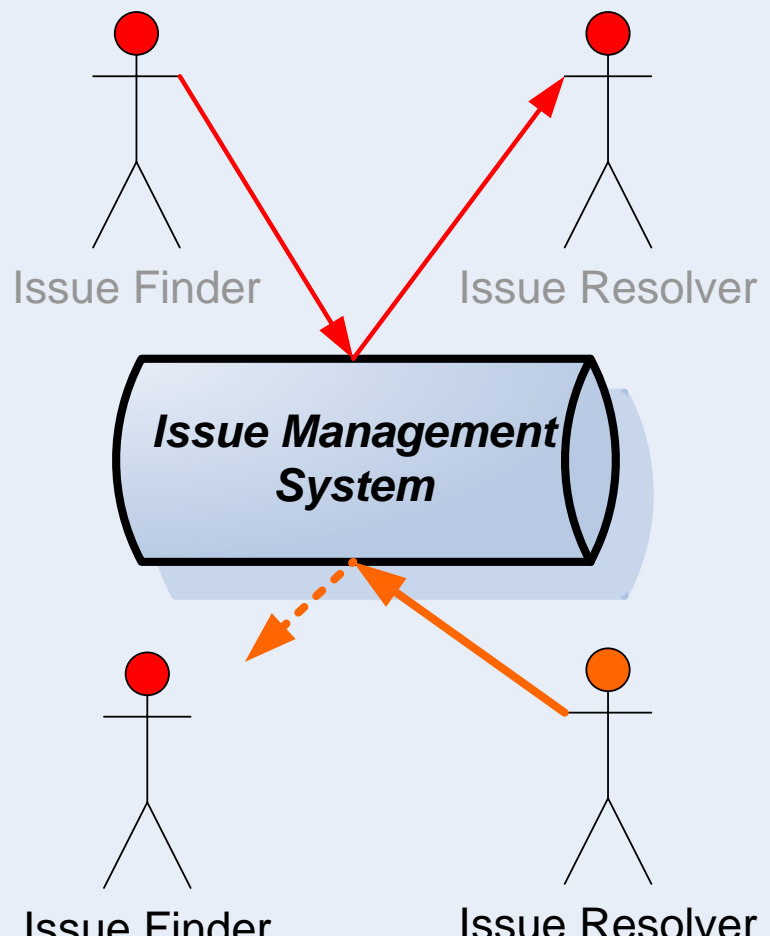
Re-Assigning Issues

Resolving an issue

Reassign open issues among staff based on skills, knowledge, experience, and workload



Resolving an issue - example



The diagram illustrates the workflow of an Issue Management System. It features four stick figures representing users: two 'Issue Finders' and two 'Issue Resolvers'. A central blue cylinder labeled 'Issue Management System' is the hub. Red arrows show an Issue Finder reporting an issue to the system, which is then assigned to an Issue Resolver. An orange arrow shows an Issue Resolver reporting a resolution back to the system, which is then notified to another Issue Finder. The text 'Issue Resolving' is written in large orange letters at the bottom.

Case 13: Cross Section Features Not Validating
Washington County, WI – 3. Data Development

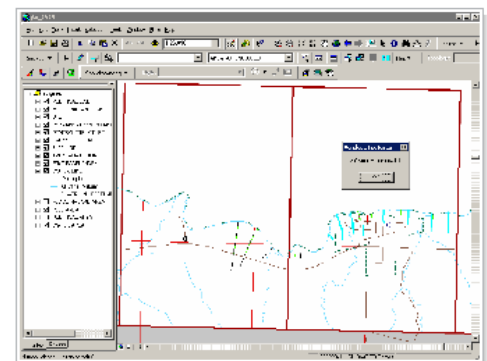
[Edit](#) [Assign](#) [Email](#)

Status: **Resolved (Fixed)** - [Add Release Note](#)
Assigned To: [Justin Kauffman \(GIS Engineer\)](#)
Category: **Bug**
Fix For: **Data Complete: 6/16/2006**
Priority: **1 – Must Fix**
Due: **6/12/2006 9:00 AM**
Estimate: (original) **1 hour**
(current) **1 hour**
(elapsed) **0 hours**
(remaining) **1 hour**

Opened by [Justin Kauffman \(GIS Engineer\)](#) 6/9/2006 (Today) 11:29 AM

1. Started JTX and GeoPop
2. Started Editing
3. Added cross section features to dataset
4. Clicked Validate Features button
5. Error Message: 8 Features invalid

What should I do?



Assigned to [Michael Grasso \(Project Manager\)](#) by [Justin Kauffman \(GIS Engineer\)](#) 6/9/2006

Resolved (Fixed) by [Michael Grasso \(Project Manager\)](#) 6/9/2006 (Today) 11:43 AM
Estimate changed from '0 hours' to '1 hour'.
Fix For changed from 'Undecided' to 'Data Complete: 6/16/2006'.
Date due changed from (No Date) to 6/12/2006 9:00 AM.

1. Click on the OK button on the Error Message box
2. Select one highlighted invalid feature
3. Click on the Validate Features button again.
4. You will see a more specific error message

It is likely an invalid attribute that you will need to fix.

Resolution example

Resolving an issue

By	Krishnappa, Prakash
Date	8/21/2004 4:49:00 PM
Response	<p>The issue is fixed by adding the validation for the user logged on against the AssignedByID. If they are same then they are allowed to delete the ASR. Also the UI is tightened further. The supervisor cannot reassign the ASR. He can only change the Status, add the status related comments & date & then save it . ALL other fields are locked. This patch is applied since there was no supervisory(ASR level) group in the security database.</p> <p>I have added the ASR level group in the security database. Changed the dependency code in AssignedSR.asp & global.asa.</p> <p>Fei check the functionality and let me know.</p>
Entity	Service Request
Type	Security
Assigned To	Fei Wang
Severity	Sev-2
Priority	Normal
Status	Open
Resolution	Fixed

Resolving an Issue – Required Information

Resolving an issue

- Estimate
- Resolution
- Status
- Note
- Assigned To
- Resolved by
- Date/Time Resolved
- Related Issues

Estimate

Resolving an issue

- ✓ **The developer should estimate how long it will take to resolve it and fill in the estimate**
 - ***Adjust the estimate for big issues that still require work, this enables tracking how much work remains***
- ✓ **Some issue tracking systems track the “Original Estimate”, and provide other fields, such as:**
 - **“Current Estimate” for re-estimating the remaining work**
 - **“Elapsed Work” to enable the developer to show how much time has been expended**
 - **“Time Remaining” can be calculated by subtracting elapsed from current estimate**

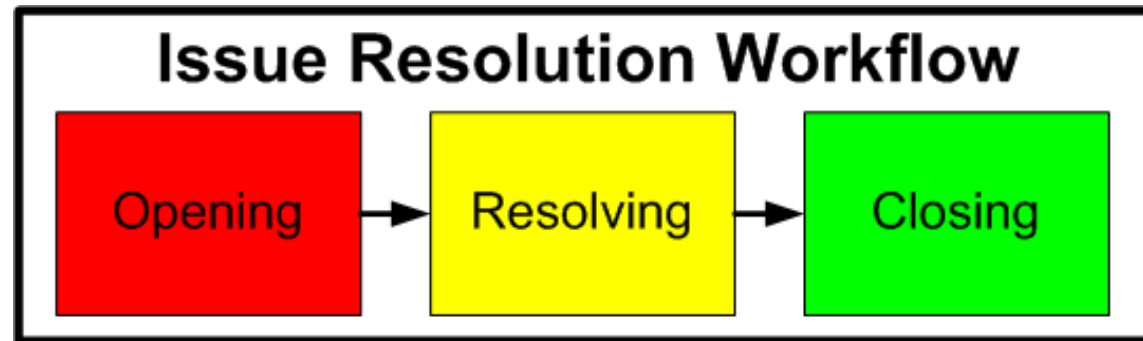
- Bugs can be resolved as:
 - Fixed
 - Not Reproducible
 - Duplicate
 - Postponed
 - Won't Fix
 - By Design

- New feature requests can be resolved as:
 - Implemented
 - Postponed
 - Won't Implement
 - Already Exists

Issue Resolution Workflow

There are 3 phases in the issue resolution workflow:

1. Opening
2. Resolving
3. **Closing**



Closing an Issue

If the resolution is valid, the issue's Status is set = to "Closed"

Case 7: Redline Feature cannot be saved
ArcIMS Data Viewer – Misc

← Previous Next →
Edit Reopen

Status: **Closed: Resolved (Fixed)** - [Edit Release Notes](#)
Assigned To: **CLOSED**
Category: **Bug**
Fix For: **Undecided**
Priority: **3 – Must Fix - Low**
Correspondent: rraad@ch2m.com

Opened by [Fei Wang](#) 3/29/2006 3:18 PM

Reproduce steps:
1) go to <http://imserver/wilmington>
2) zoom in the map so that you can see buildings and pipes
3) click on "redline tool"
4) digitize a polygon
5) click on Save button on Redling frame
6) type in Title, Description, Name, then click on Save Session
7) an error message shows up: "Error: adding session information - details: Operation must use an updateable query"

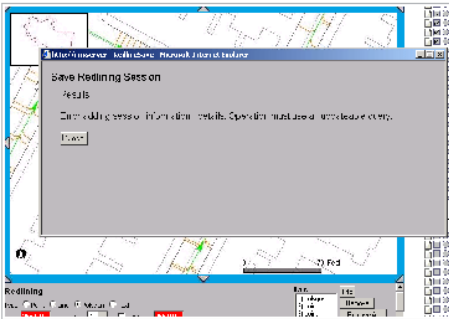
what is an updateable query?
I didn't see any definition in their User Guide document

Assigned to [David Lanter](#) by [Fei Wang](#) 3/29/2006 3:18 PM

Assigned to [Fei Wang](#) by [David Lanter](#) 3/29/2006 5:10 PM

Please attach screen capture of error message.

Edited by [Fei Wang](#) 3/29/2006 5:14 PM



Assigned to [David Lanter](#) by [Fei Wang](#) 3/29/2006 5:15 PM
Version changed from "" to 'Beta 1'.

screenshot is attached

Assigned to [Rami Raad](#) by [David Lanter](#) 3/29/2006 7:58 PM

This bug's for you.

Emailed by [Rami Raad](#) 3/30/2006 6:54 AM

From: "Wilmington_GIS-IssueTracker" <mail@cdm.si-dios.info>
Date: Thu, 30 Mar 2006 06:54:04 -0500
To: rraad@ch2m.com
Subject: (Case 7) Redline Feature cannot be saved

IMS issue initiated by CDM

Resolved (Fixed) by [Rami Raad](#) 3/31/2006 7:07 AM
this was resolved-it was a permission issue to access certain folders on IMS server.

Closed by [Fei Wang](#) 4/4/2006 2:25 PM
it works now.

Closing an Issue – Required Information

- Resolution
- Status
- Closed by
- Date/Time Closed

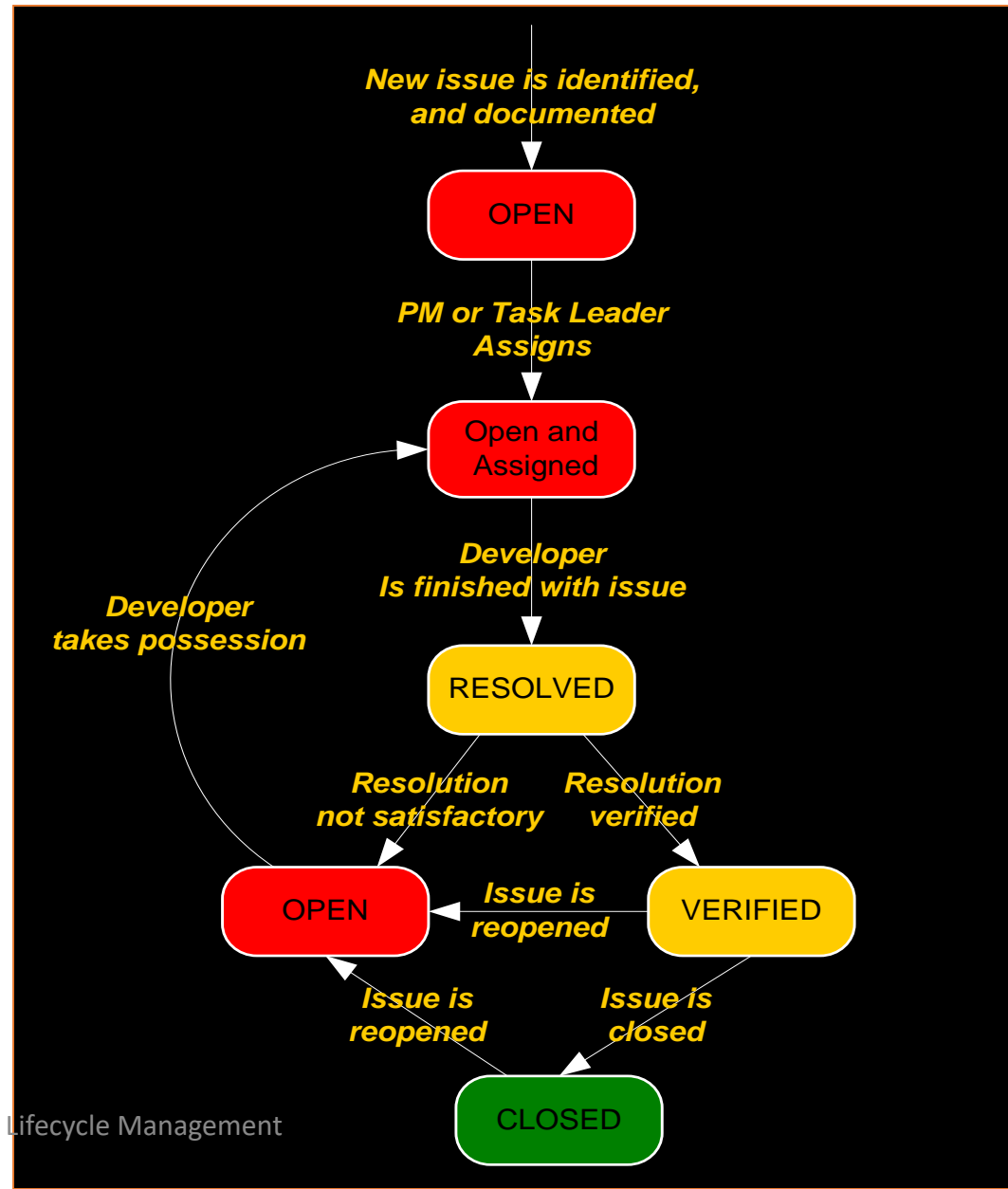
Reopening an Issue

Closing an issue

If an issue resolution is found to be incorrect (e.g. the bug is found to be not fixed):

- Status is changed from “Resolved” to “Open”
- Resolution is changed from “Fixed” to null
- Note is updated to let the Developer know that the issue is still open, providing any additional guidance for reproducing it
- Issue is reassigned by changing Assigned To and assigning it back to the Developer for resolution

Issue Resolution Workflow Summary



Standard Operating Procedures

✓ Issue Resolution Workflow

- **Triage**
- Lessons Learned

Triage

The process of sorting, prioritizing, and selecting which issues to allocate resources to, and which to postpone or avoid based on decisions concerning:

1. Where funds are most needed and can be best used
2. Which development risks are acceptable, and which should be avoided based on abilities of the development team, remaining budget, and time left in the schedule

A Triage-based View of Project Schedule

- **Pre-Triage**

- From “Project Kickoff” project milestone through “Application Development 50% Complete” milestone
- Scoped features designed and prototyped, and Developers implementing critical core features

- **Early Triage**

- From “Application Development 50% Complete” milestone to “Code Complete” milestone
- Testing begins and issues are recorded, prioritized, and resolved.

- **Late Triage**

- From “Code Complete” milestone through Alpha, Beta, and “Final Release and Delivery” of the application
- No new features worked on

Triage – Analyzing Issues

Issues from issue tracking systems can be analyzed in reports, Excel Pivot Tables and Charts to assess application stability and project status

The image shows a screenshot of Microsoft Excel with a PivotTable and PivotChart Report. The PivotTable is set up to analyze issue data, with 'Entity' as the row labels and 'Severity' as the column labels. The PivotTable shows the following data:

Entity	Sev-1	Sev-2	Sev-3	Grand Total
Asset/System	26	13	12	51
Dispatcher			1	1
Inventory	1		3	4
Map Layers		1	2	3
Service Request		1	1	2
Grand Total	27	15	19	61

The PivotTable Field List on the right shows the following fields:

- Issue #
- Issue
- Issue Description
- Status
- Priority
- Submitted By
- Date Submitted
- Originally Assigned To
- Currently Assigned To
- Last Updated By
- Last Update Date
- Entity
- Severity
- Resolution
- Release

Triage – Scheduling a Release

Examine the pivot table of Open issues below. Assume that all the areas (“Entity” is a synonym for “Area”) have been tested, and Entity = “ALL” identifies issues affecting all areas

1. Is this application ready for delivery to the client?
2. Which subsystems are ready for delivery?
3. Formulate a strategy for quickly achieving a first release of this application.

Count of Issue #	Severity				
Entity	Sev-1	Sev-2	Sev-3	Sev-4	Grand Total
Admin	5	3	4	1	13
ALL	1	9	4	1	15
Asset/System	38	77	39		154
Citizen Complaint	3	16	10	1	30
Dispatcher	1	1	3	2	7
Inspection		1			1
Inventory	4	8	4		16
Map Layers	1	1	3		5
Project		1			1
Report			6		6
Service Request	22	80	33	9	144
Spill Response			1		1
ToDo	2	4	1	1	8
WorkOrder	2	7			9
Grand Total	79	208	108	15	410

Triage – Managing Resources

- Examine the co-variation of issues by their severities and assignments to developers
 - Keep the focus on Severity 1 and Severity 2 issues
 - Look for overloaded developers
 - Determine if the resolution of issues can be better spread across the development team
 - Reschedule issues for later releases, if possible
 - Determine if additional resources are needed
 - Proactively forecast schedule slips

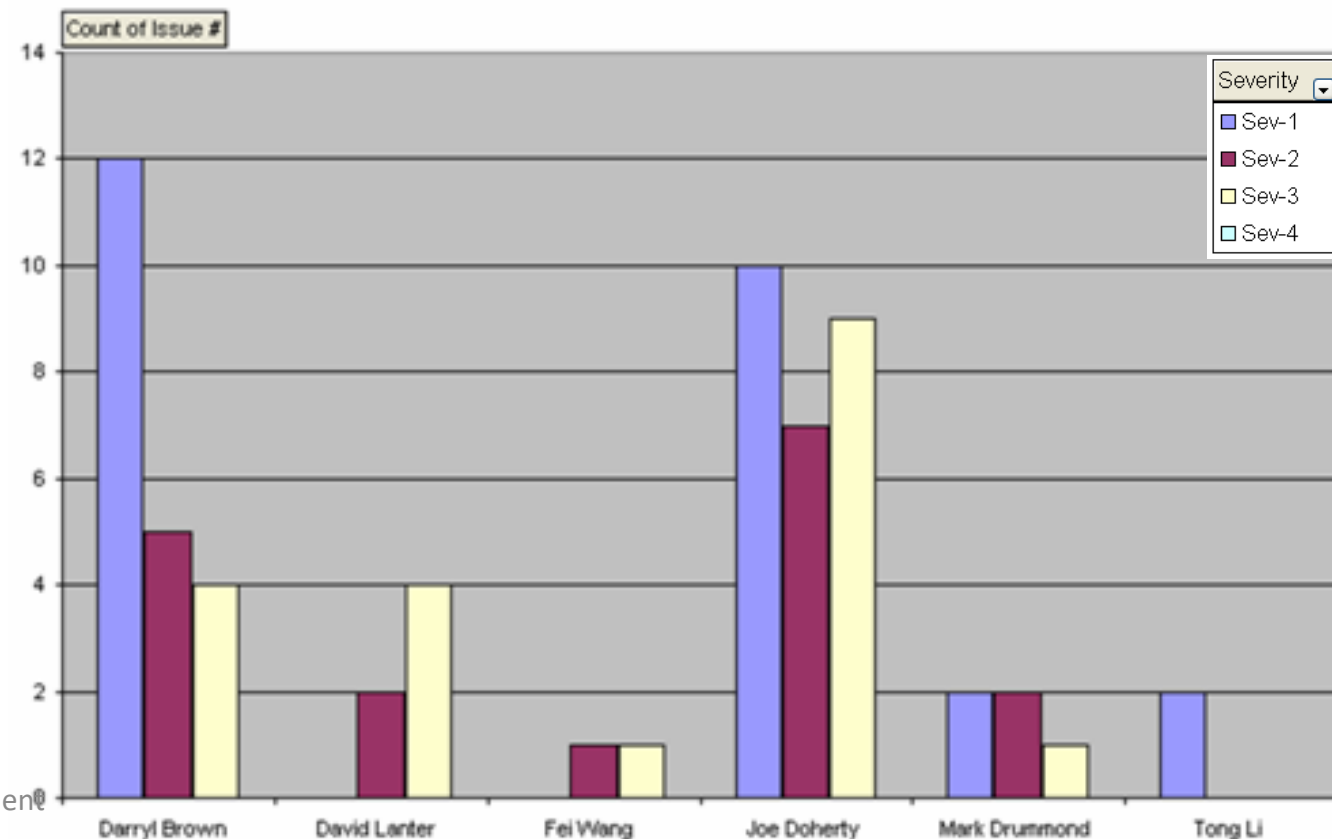
Example - Examine the graph in the email below

Who looks like they may be overloaded?

Formulate a strategy for managing the staff to handle the overload.

From: Lanter, David
Sent: Tuesday, February 01, 2005 2:09 PM
To: Doherty, Joseph; Brown, Darryl; Wang, Fei; Li, Tong; Drummond, Mark; Nehke, Seth
Subject: FDOT Issues...

Here's the distribution of R3B-1 Issues in the issue tracking system. Please make sure to periodically check the issue tracker, query for Release = R3B-1, and Status not= Closed, and clear out the bugs assigned to you.



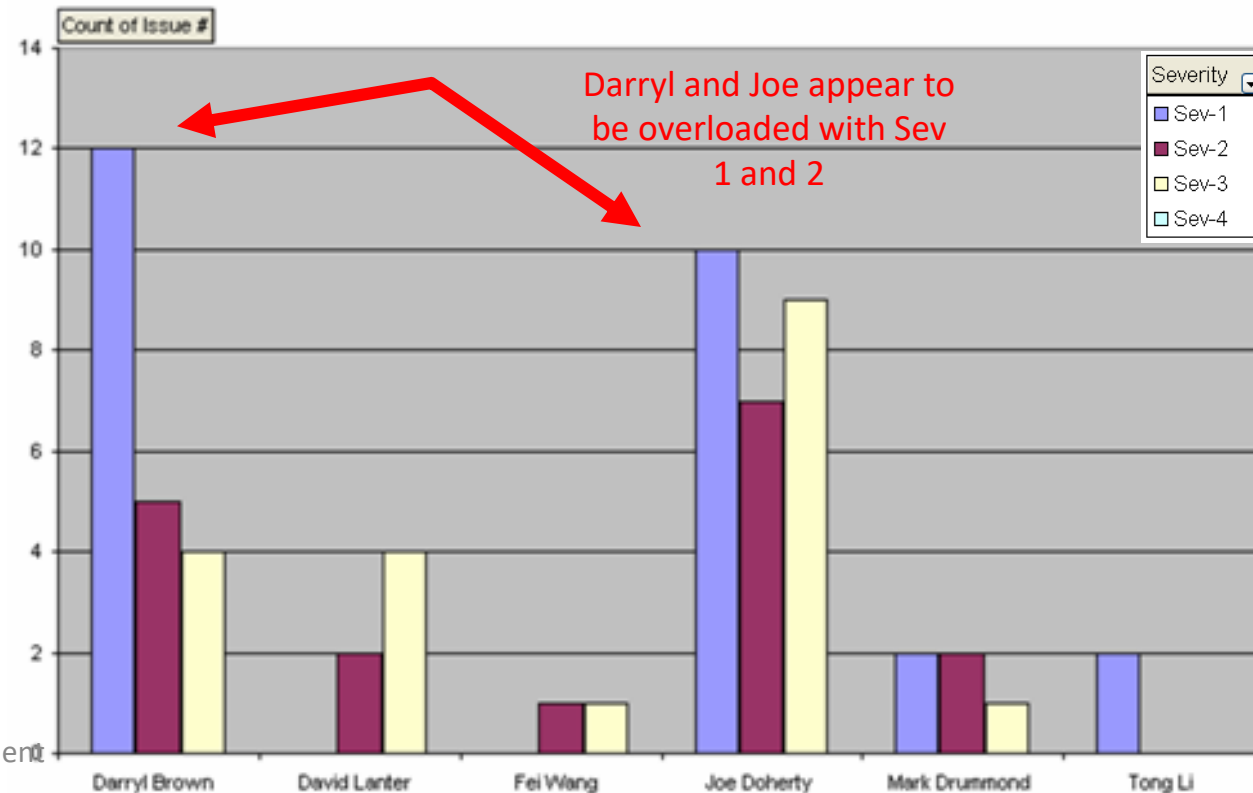
Example - Examine the graph in the email below

Who looks like they may be overloaded?

Formulate a strategy for managing the staff to handle the overload.

From: Lanter, David
Sent: Tuesday, February 01, 2005 2:09 PM
To: Doherty, Joseph; Brown, Darryl; Wang, Fei; Li, Tong; Drummond, Mark; Nehrlke, Seth
Subject: FDOT Issues...

Here's the distribution of R3B-1 Issues in the issue tracking system. Please make sure to periodically check the issue tracker, query for Release = R3B-1, and Status not= Closed, and clear out the bugs assigned to you.

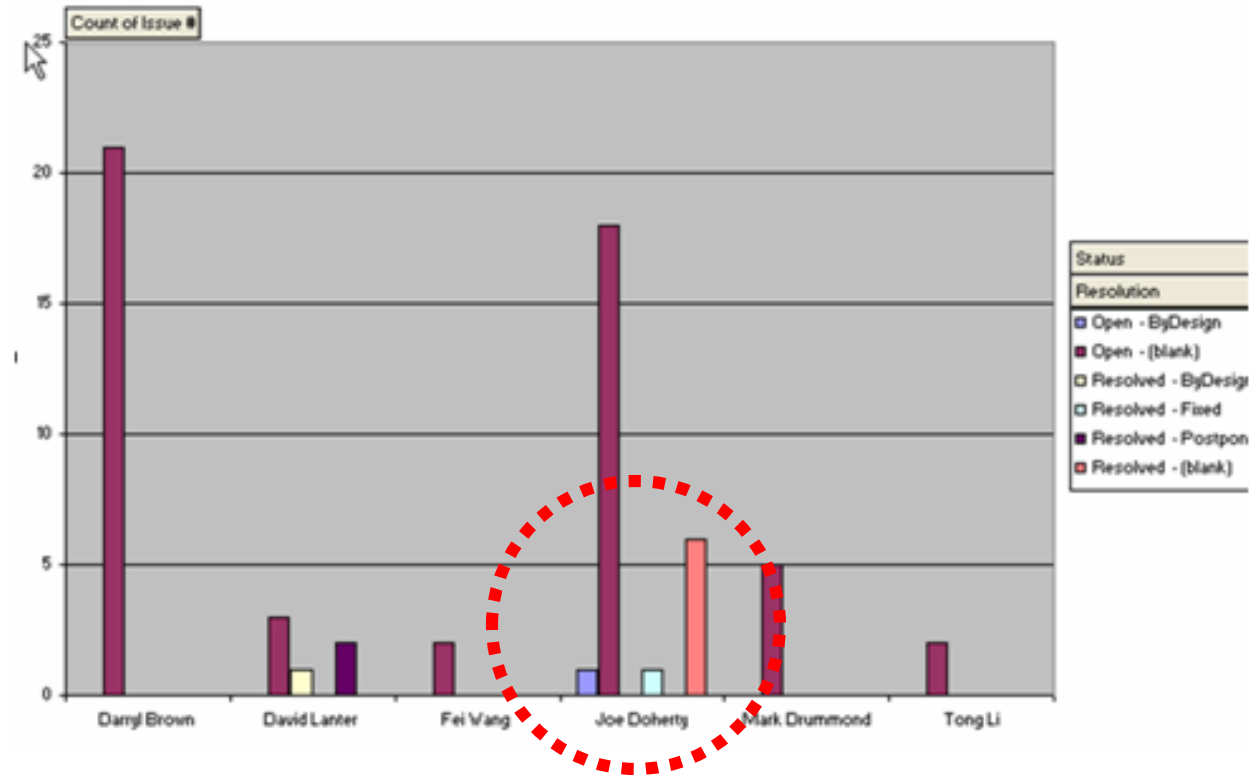


Good Issue Management

From: Lanter, David
Sent: Tuesday, February 01, 2005 2:09 PM
To: Doherty, Joseph;
Subject: FDOT Issues...

Joe, the next chart suggests that you are resolving issues - but when you resolve them you also need to assign them to someone (likely Mark or Tong) to confirm the resolution and close them.

Requires a bit of policing and guidance, until everyone is familiar with the issue resolution workflow and what they need to do within it



As a reminder:

Work through the bugs by severity, Sev-1's first, then Sev-2's, then Sev-3's.
If the issue is assigned to you, resolve it, and reassign back to the finder (usually Mark, Tong, or Seth) for them to agree it's been resolved and then they can close it.
If you found the issue, and it is now re-assigned back to you, check to see that it has been resolved and close it.

If you think the issue should be postponed or if there is any other complication we should discuss, then assign the issue to me and I'll follow up.

Triage – Managing Final Delivery

- Resolved issues must be validated and closed
- Remaining Open issues are
 1. Fixed if they are truly Severity 1 or Severity 2, or postponed if Severity-3 or Severity-4
 2. Closed
- All issues must be closed before an application development project can be closed out

Agenda

- ✓ Installation & change management
- ✓ Software applications testing
- ✓ Quality characteristics
- ✓ Error detection techniques
- ✓ Entity inspection
- ✓ Evaluating datasets
- ✓ Issue tracking

