

## Improving IV&V Techniques Through the Analysis of Project Anomalies: LINKER - preliminary report

**Tim Menzies**

Lane Department of Computer Science and Electrical Engineering, West Virginia University, USA

[tim@menzies.us](mailto:tim@menzies.us)

<http://menzies.us>

**Abstract** This project is in two parts. The *second part* will try to combine two (or more) of the IV&V data sources into an active monitoring framework where data collected during an active IV&V project will trigger an alert if a project becomes unusual” (and defining “unusual” is one of the goals of this project).

Before we can generalize between sources, we need to study each source in isolation to determine its strengths and weaknesses. Hence, the *first part* of this project aims to gain experience with the various IV&V data sources available to researchers like myself; i.e.

- SILAP, from the IV&V planning and scoping team;
- James Dabney’s Bayes networks that describe the IV&V business practices of the L3 IV&V contractor;
- The PITS issue tracking data;
- The LINKER database project that intends to join PITS to other data sources;
- Balanced score card strategy maps from NASA Langley.
- and the COCOMO data sets from JPL.

To date, this project has resulted in:

- A preliminary report described what had been learned from the SILAP data. A ranking was offered on the most common IV&V work-breakdown structure (WBS) activities. This ranking can be used for (e.g.) identifying what WBS tasks would benefit most from optimization.
- A preliminary report Dabney’s Bayes networks. On a limited case study, it was shown that Bayes nets and treatment learning could generate parsimonious explanations for project events.
- A preliminary report on text mining from the PITS issues tracking database that generated an expert system which audited a test engineer’s proposed severity level.

This report describes LINKER, a new database at IV&V that combines data from different sources. LINKER is still under-development. Hence, unlike the prior reports, this report does not report experimental results. Rather, it discusses the benefits and drawbacks of LINKER.

---

**Credits:** This report was generated with the assistance of Wesley Deadrick (computer engineer at NASA IV&V). This research was conducted at West Virginia University under NASA sub-contract project 100005549, task 5e, award 1002193r.

**Cautions** Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government.

---

**Contents**

1 Introduction . . . . . 2  
 2 LINKER as a Research Tool . . . . . 3  
 3 LINKER for Anomaly Detection . . . . . 4  
   3.1 About Belief Networks . . . . . 4  
   3.2 Belief Networks for IV&V . . . . . 6  
   3.3 Assessing LINKER . . . . . 6  
 4 Solo Mode . . . . . 7

**List of Figures**

1 PITS . . . . . 2  
 2 LINKER . . . . . 3  
 3 A Bayes net of IV&V practices . . . . . 5  
 4 Defect introduction . . . . . 5  
 5 Defect removal (developer) . . . . . 7  
 6 Defect removal (IV&V) . . . . . 7

**1 Introduction**

We seek methods to monitor NASA projects using the database housed at IV&V. One such data base is PITS, the Project and Issue Tracking System shown in Figure 1.

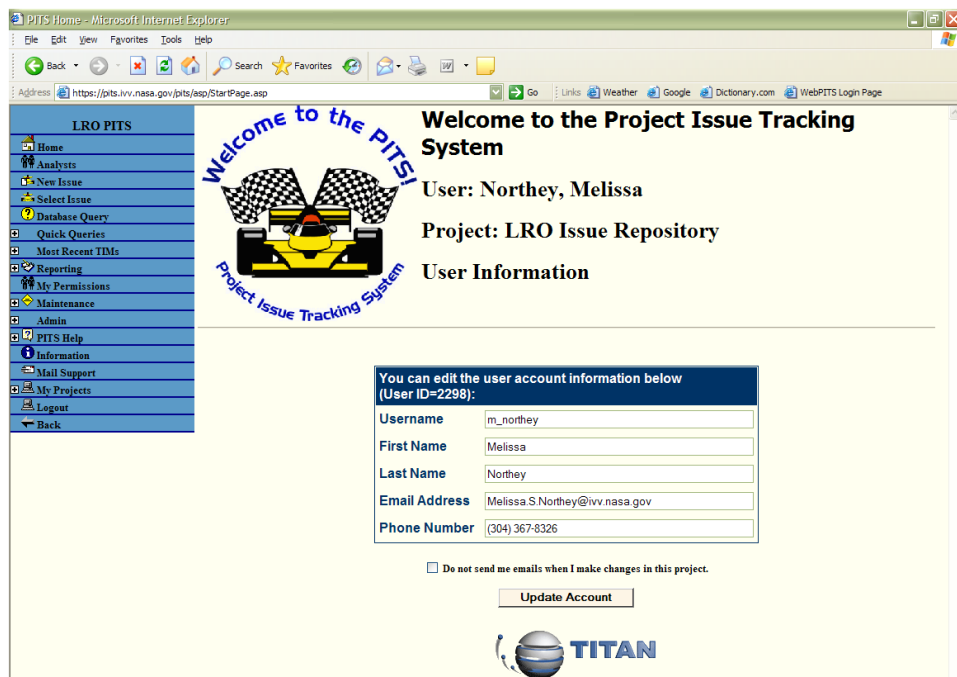
While PITS works well for queries over one project, it is a poor interface for running queries over multiple projects. Also, there is a lack of consistency in how each of the projects collects PITS issue data. While there was a set of required data fields, the majorities of those fields do not provide information in regards to the quality of the issue and are not very suitable for comparing projects.

NASA is very aware of the problems with PITS and is taking active steps to improve it. At the time of this writing, there is an on-going effort to implement a mandatory data set in each IV&V project database to support IV&V effectiveness metrics.

As part of the effective metrics work, Wesley Deadrick is leading the LINKER project that aims to act as a centralized query system for IV&V data. LINKER will have read-only access to the other IV&V databases and offer a centralized query facility that can reach into the other databases. Internally, LINKER works by first importing data from other IV&V databases (see Figure 2). Initially, LINKER is only targeting PITS but, when other databases become available, Deadrick’s team would integrate them as well. Once imported, LINKER can then explore data from multiple projects.

This report is an audit of LINKER. Specially, we will ask “how good is LINKER?”:

- As a standard research tool. Several drawbacks with the current version of LINKER will be discussed.



**Fig. 1** The login screen to the PITS database.



Fig. 2 LINKER, accessing data from other IV&V databases.

- For detecting anomalies in active IV&V projects. Our analysis shows that LINKER stores only a minority of the data points needed for this anomaly detector.

However, the point of this report is that LINKER is not some static, already written, set-in-stone piece of software. Rather, it is the focus of an active and motivated development team. The goal of this report is to offer insight to that team into how best modify the current version of LINKER in order to enable its use in the anomaly detection process. To that end, this report:

- Makes 14 specific recommendations about extending the current version of LINKER.
- Discusses what could be done if *none* of those recommendations are implemented- see §4 about *solo mode*.

## 2 LINKER as a Research Tool

For the purposes of this report, “researchers” means *either* NASA civil servants looking for IV&V-specific issues *or* university researchers seeking general methods for improving software engineering. This section discusses the merits of LINKER for use by researchers.

LINKER was designed to overcome certain limitations with PITS. The current interface of PITS can only run queries against a single project. There is no way to select multiple projects or to save a query generated on one screen for one project and to use it on another project.

Once projects are selected, LINKER offers a set of canned queries for report generation. While such canned queries are a useful way to introduce novices to a system, they cannot capture the space of possible research queries:

- The merits of canned-vs-arbitrary queries was demonstrated recently at a review for a WVU university initiative (December 13, 2006).
- Two WVU researchers discussed their work on a two year cycle of “get data from project” and “consider what to ask for next”.
- They found that every data dump from the projects offered hints on what other information might be most informative.
- In fact, to complete their research, they needed to do large scales dumps of the data followed by intricate, special purpose queries.

The lesson of the above is clear:

**Recommendation 1** *LINKER should offer query support for both novice users (who need a user-friendly GUI controlling some canned queries) as well as more experienced users (who need to access the raw SQL queries).*

**Recommendation 2** *LINKER should be augmented with SQL query tools for power-users (e.g. string matching on partial words).*

**Recommendation 3** *LINKER users should be able to create, read, update, and delete a library of arbitrary queries over IV&V projects.*

(Regarding **Recommendation 2**: Deadrick comments that the LINKER team is currently designing a “configurable query tool” which allows users to construct queries on the fly which span multiple projects. Due to inconsistencies in the PITS database, this is proving to be a challenging task.)

Quite apart from the technical issues, PITS data must be kept secure. Access to project data can be controlled on a

project-by-project basis and accessing all data on all projects might involve an elaborate and extended administrative task of collecting the required permissions.

**Recommendation 4** *The permissions policy needs to be clarified. A trade-off between synonyms and automatic access might be appropriate. E.g. permissions to access all data automatically enables replace all domain terminology with anonymous synonyms.*

One solution to this security issue would be to filter all data from PITS via a set of *synonym tables* that obfuscate the specific project data, perhaps at increasing levels of security.

**Recommendation 5** *The LINKER sysadmin should maintain tables of synonyms for each unique word that appears on LINKER screens (either as fixed text on the screens or text generated from queries in result fields).*

(Regarding **Recommendation 5**: Deadrick comments that a synonym table for all LINKER fields might be overly-resource expensive. However, if synonyms are restricted to just the fields for “issue counts”, “severities”, “subsystem”, “WBS activities” and “tasks”, “phase found”, and “phase introduced”, then these synonyms could become quite simple.)

**Recommendation 6** *LINKER users should be assigned a security access code by the sysadmin. This code selects a synonym table and that table should be used to filter all the output on every LINKER screen.*

**Recommendation 7** *One synonym table should be the identity table where each word maps to itself.*

**Recommendation 8** *Another synonym table should be the sanitized table where each word maps to some meaningless identifier (e.g. x00001, x00002, etc).*

**Recommendation 9** *Users with the greatest security clearance, should get assigned the identity table.*

**Recommendation 10** *Users with the least security clearance, should get assigned the sanitized table.*

Once the synonyms are in place, then LINKER can be made broadly available:

**Recommendation 11** *LINKER should be freely available to the broader community outside of the facility and the default account enables the anonymous synonyms.*

### 3 LINKER for Anomaly Detection

Having assessed the current version of LINKER, this report now discusses how it might be extended to support anomaly detection. Anomaly detection is at least a three-stage process:

- In the first stage, some *expectation* is developed for observable values. A requirement for this first stage to work is some *relevant background knowledge* of the domain. Our current best candidate for *background knowledge* are the Bayes networks developed by James Dabney et.al. [1] describing the practices of an IV&V contractor. One advantage of Bayes nets are that they can handle uncertain information. Also, their internal distributions can be incrementally refined based on updated domain information. That is, these nets can be kept *relevant* to changing conditions, using an automatic process.
- In the second stage, some *observables* are collected from the domain. A requirement for this second stage is access to an *active data source*.
- In the third state, the probability of the observables is computed using the expectations. If that probability is low, then an anomaly is detected. Of these stages, this stage is the simplest to achieve. Decades of AI research has resulted in Bayes classifier technology [2,3], that compute the expected likelihood of prior projects, and detects when a new example is an outlier from the past [4].

Since we have some confidence in stages one and three, this report will focus on stage two. We ask the questions “is LINKER a candidate *active data source*?” We will seek holes in LINKER’s schema by “asking” it what it can say about the background knowledge represented in Dabney’s Bayes nets. Therefore, to assess LINKER we first need to understand Dabney’s belief nets.

#### 3.1 About Belief Networks

A belief network (also known as a Bayesian network or a Bayes net) is a directed acyclic graph of

- nodes representing variables (nodes can represent any kind of variable, be it a measured parameter, a latent variable or an hypothesis);
- arcs representing probabilistic dependency relations among the variables and local probability distributions for each variable given values of its parents.

If there is an arc from node *A* to another node *B*, then variable *B* depends directly on variable *A*, and *A* is called a *parent* of *B*. If there are *n* variables, then for each variable  $X_i$  ( $1 \leq i \leq n$ ), the set of parent variables is denoted by  $parents(X_i)$  then the joint distribution of the variables is product of the local distributions

$$Pr(X_1, \dots, X_n) = \prod_{i=1}^n Pr(X_i | parents(X_i))$$

If  $X_i$  has no parents, its local probability distribution is said to be *unconditional*, otherwise it is *conditional*. If the variable represented by a node is *observed*, then the node is said to be an *evidence* node.

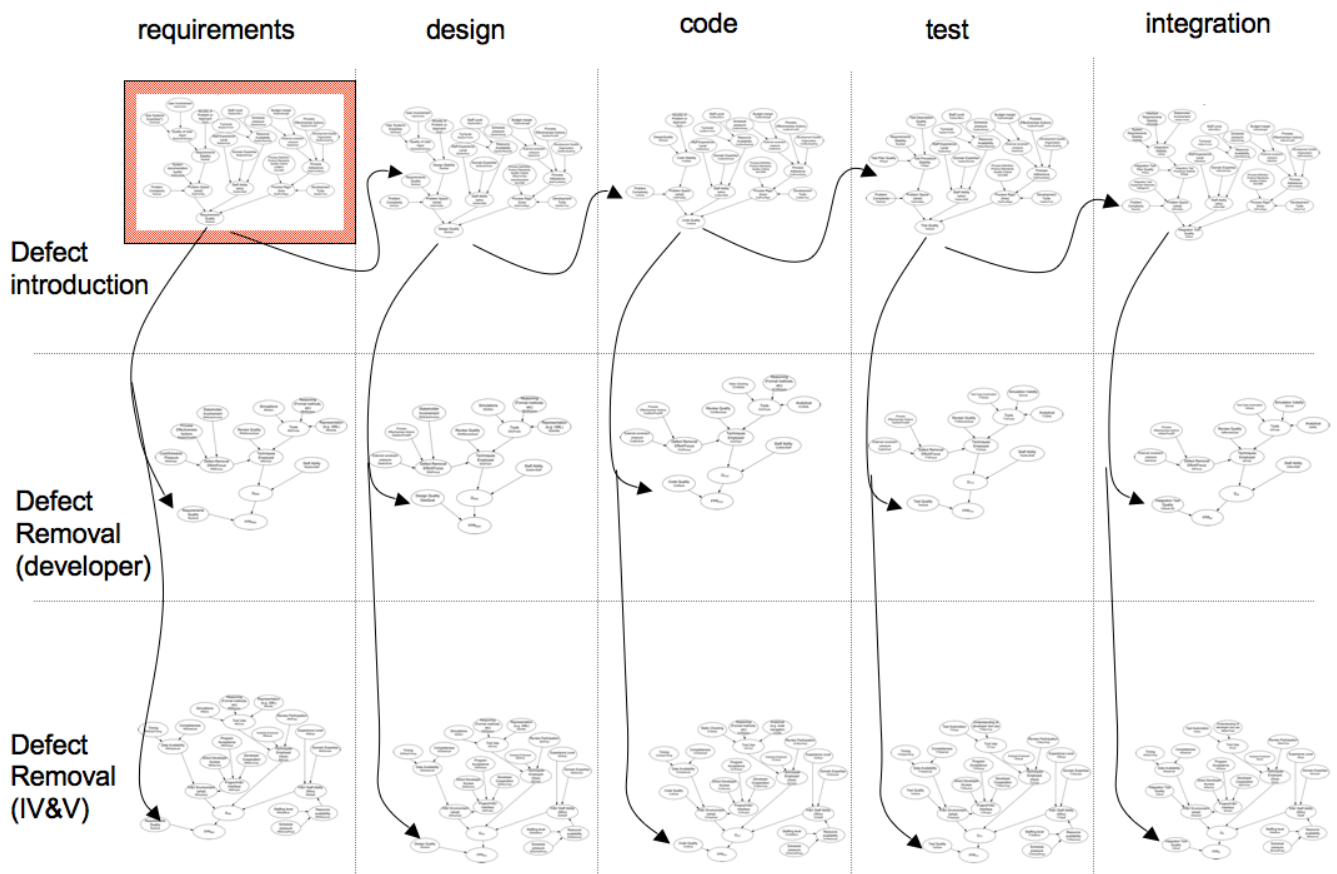


Fig. 3 Dabney’s Bayes net describing IV&V practices. The red square, top left, is expanded in Figure 4.

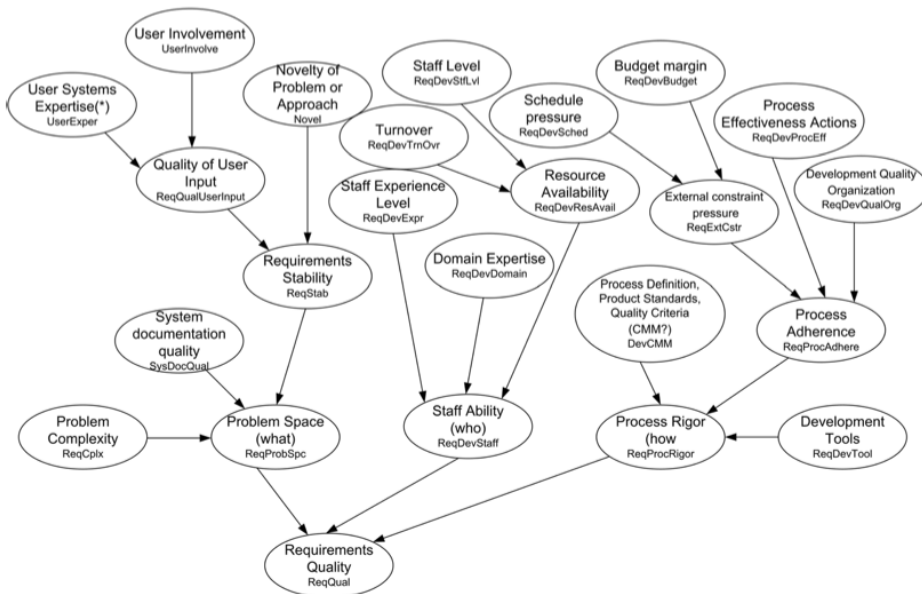


Fig. 4 The “defect introduction” Bayes sub-net shown in a red box, top-left of Figure 3.

### 3.2 Belief Networks for IV&V

Using this representation, Dabney described the IV&V practices of a NASA contractor. That description comprised:

- The Bayes net of Figure 3;
- Distributions for the inputs to the network for three different NASA projects.

Figure 3 may appear complex, but its structure is actually quite simple:

- The *rows* represent the actions of developers and IV&V analysts working on NASA software. Row one shows developers building software and, inadvertently, introducing defects. Developers are well aware of their shortcomings so, in row two, they perform numerous defect removal activities. In order to ensure software quality, NASA runs an IV&V team which (in row three) also removes defects.
- The *columns* represent a standard NASA software development project:

*requirements* → *design* → *code* → *test* → *integration*

- The *horizontal thick arrows* in the diagram model defects generated at each phase leaking into subsequent phases.
- The *vertical thick arrows* model defects being generated, then removed by either the developers or the IV&V team.

In terms of understanding Dabney's networks, the essential feature of Figure 3 is that each column stores essentially the same structure, with minor differences; e.g. occasionally a node might have three parents, not two. Hence, reasoning about much of the system can be simply achieved via a closer examination of any of the columns.

### 3.3 Assessing LINKER

Having described all that, the next question is “how many nodes in Figure 4 and Figure 5 and Figure 6 can we observe in actual projects?”. In this regard, the “Techniques Employed” subset (top right of Figure 5) is particularly insightful. If details of tool use were routinely stored in the IV&V project monitoring databases, then those fields might never be filled in. For contractual reasons, the IV&V database does not include information about the specific tools used by IV&V contractors. The general rule is that NASA civil servants describe “what” is to be done and the contractors are free to decide “how” to do it, using their available resources.

Similarly, the contractors might object to offering data on the “Staff Ability” node (middle, left hand side, Figure 5) since they might require offering too much information to the civil servants.

In fact, a detailed review of Figure 4 and Figure 5 and Figure 6 by Menzies and Deadrick showed that only a minority of the nodes were directly observable (either using LINKER or any other known candidate source). In the defect introduction subnet of Figure 4:

- *System Document Quality* might be accessible via the new Cukic WVU University initiative;

- *Requirements Stability* might be accessible via a configuration management tool that recorded the rate of change in the requirements documents;
- *Novelty of Problem Approach* could be read from the SILAP planning and scoping database or a COCOMO description of the project;
- *Staff Experience Level* might be accessible if contractors could release the years of service of the contractors assigned to this IV&V task;
- *Schedule Pressure* could be read from a COCOMO description of the project;

In the developer defect removal subset of Figure 5:

- *Review Quality* might be accessible via LINKER and the number of issues found;

In the IV&V defect removal subset of Figure 6:

- *Data Availability* might be accessible from knowledge of the documents currently in-house at IV&V (for example, using “phase alignment”).
- *Program Acceptance* can be accessed in PITS/LINKER by recording how often issues were accepted, disputed, withdrawn, etc.
- *Experience Level* might be accessible if contractors could release the years of service of the contractors assigned to this IV&V task;
- *Staffing Level* might be accessible from the Planning & Scoping databases;
- *Schedule Pressure* could be read from a COCOMO description of the project;

Recall from the above discussion that Figure 3 contains five columns, each of which holds the 63 nodes (plus or minus a few) shown in Figure 4, Figure 5 and Figure 6. Assuming the above pattern repeats across the five columns, then Menzies and Deadrick review of all the data sources from all the current IV&V projects shows that we might be able to collect for  $5 \times 11 = 55$  of the  $5 \times 63 = 315$  nodes in Figure 3 ( $\frac{55}{315} \approx 17\%$ ) of the nodes.

The good news is that 17% might be enough. Bayes nets are well-defined mathematical entities that come with auto-update tools. Given background expectations of these nodes, small drips of data (say, 17%) can be enough to calibrate the nets and let them make sensible predictions about expected and anomalous behavior.

The bad news is that the current version of LINKER can only supply data on  $5 \times \frac{2}{11}$  of the nodes. To obtain the remaining data, LINKER's current functionality would have to be extended to collect data from more inputs.

**Recommendation 12** *LINKER should connect to the planning & scoping team's database of SILAP-ed software elements.*

**Recommendation 13** *LINKER should have screens to enter COCOMO data (specialized for COCOMO-I or COCOMO-II inputs- see Figures 12&15 of [4]).*

<sup>1</sup> *Review Quality and Program Acceptance*

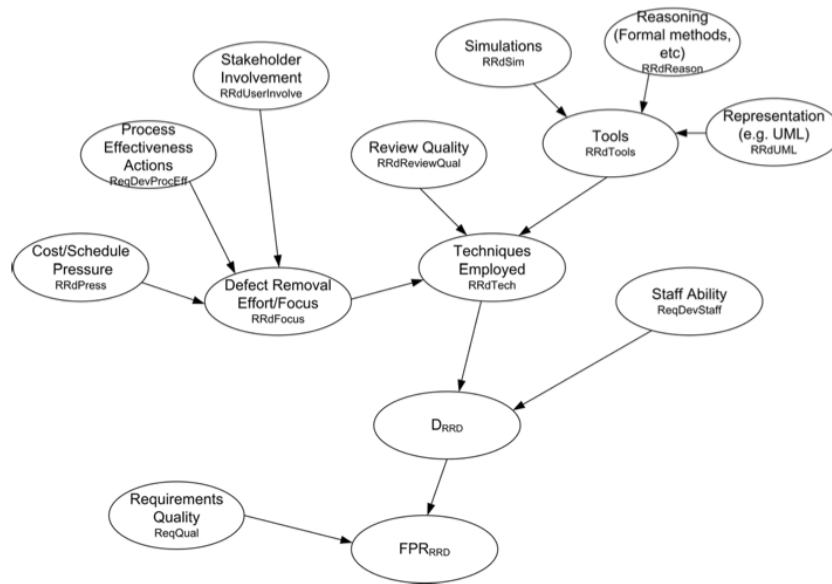


Fig. 5 The developer “defect removal” Bayes sub-net (middle row of Figure 3).

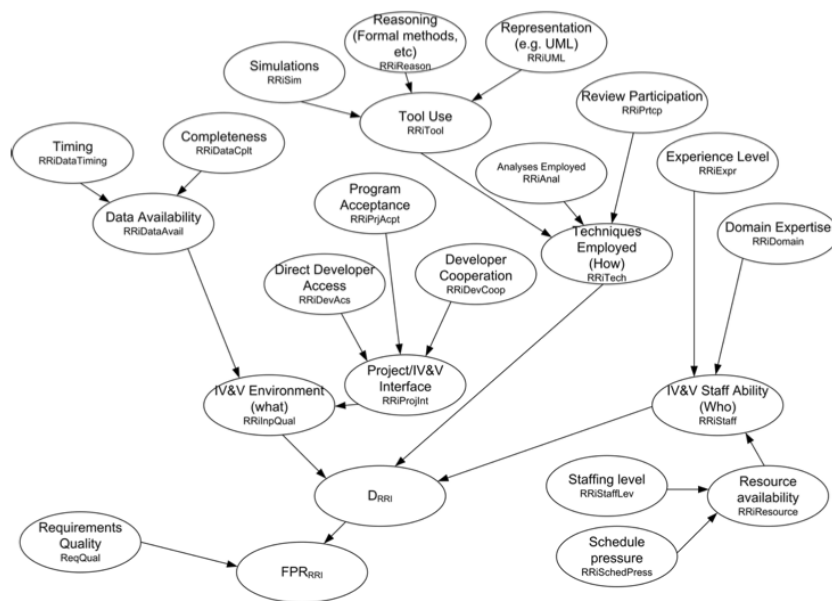


Fig. 6 The IV&V “defect removal” Bayes sub-net (bottom row of Figure 3).

More generally, there are too many “island databases” at IV&V and they need to be linked together:

**Recommendation 14** *LINKER should be extensible so it can import data from new data sources; e.g. the Cukic project on evaluating IV&V products or Galaxy Global’s MDP databases.*

**4 Solo Mode**

Every programmatic change consumes resources. If the recommendations listed above cannot be make, what is the fate of this anomaly detection project? Can it proceed?

The answer is “yes”, but in a modified form. Currently, the concept of operations is that the Bayes nets are wired into the projects via LINKER. In that mode, the nets are automatically triggered by any update to the project information. We call this the “integrated mode”.

In an alternate “solo mode”, the anomaly detector is a stand-alone application and the user is responsible for manually entering in the node data. In this “solo mode”, the NASA project manager is the API that connects the anomaly detector to the projects.

While the solo mode is not the preferred option, is it possible and practical:

- Recall from the above that Bayes nets can work from small amounts of imprecise data. If once a month, the manager spent 20 minutes with the tool entering in the available data, then they could sufficient for anomalies to be detected.
- Under the hood, the data entered by the manager could be stored away in a database, so results from different projects could be compared and combined.
- Data miners could run over the models looking for the *least* number of nodes that the user has to enter in order to describe the current tactical situation (this would reduce the data entry effort of the user).

In fact, solo mode could be used to bootstrap integrated mode. If the recommendations made here are too expensive to be performed in the short term, then a successful track record with solo mode could be used to lever funds to make the required changes to LINKER which, in turn, would enable integrated mode.

## References

1. J. B. Dabney, G. Barber, and D. Ohi. Predicting software defect function point ratios using a bayesian belief network. In *Proceedings of the PROMISE workshop, 2006*, 2006. Available from <http://promisedata.org/pdf/phil2006DabneyBarberOhi.pdf>.
2. P. Domingos and M. J. Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3):103–130, 1997.
3. G. John and P. Langley. Estimating continuous distributions in bayesian classifiers. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence Montreal, Quebec: Morgan Kaufmann*, pages 338–345, 1995. Available from <http://citeseer.ist.psu.edu/john95estimating.html>.
4. T. Menzies, D. Allen, and A. Orrego. Bayesian anomaly detection (bad v1.0). In *Proceedings of the Machine Learning Algorithms for Surveillance and Event Detection Workshop, ICML'06*, 2006. Available from <http://menzies.us/pdf/06bad.pdf>.