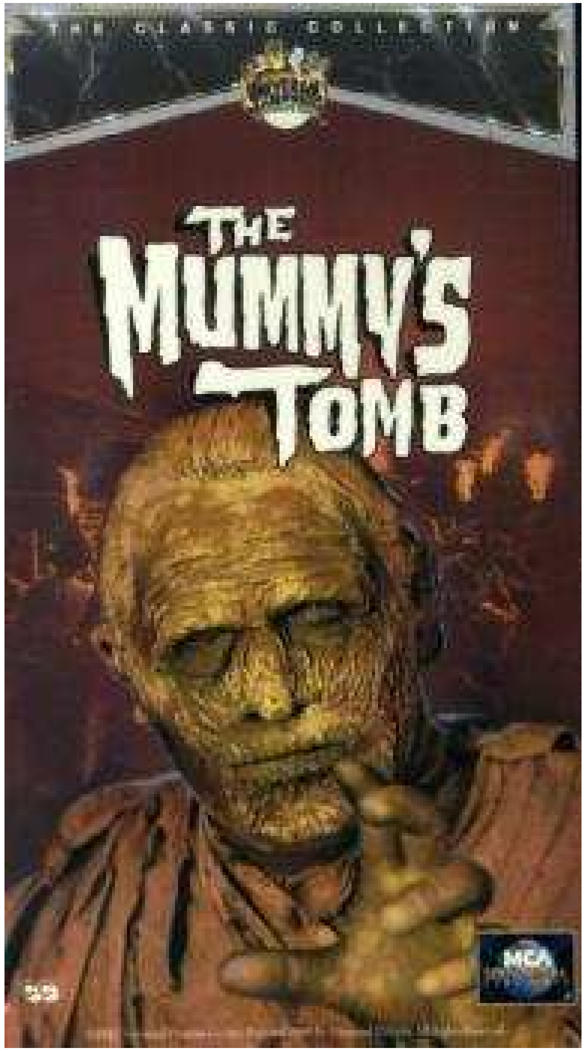
“Telling More“: a 2006 NASA SARP project (Technical Briefing)

Tim Menzies
Lane Department of Computer Science and Electrical Engineering
West Virginia University
tim@menzies.us

July 17, 2006

Introduction: Use it or Lose it (and this is the last "business-level" slide)



- NASA data = active repository or data tomb?
  - write once;
  - read never;
  - buried;
  - doomed;
- If an organization spends millions of dollars on data collection and archiving...
  - It should spend tens of thousands (at least) in analyzing that data.
- NASA IV&V is in a unique position to review and comment on much of the NASA software enterprise.
  - We *see* more, but what have we *learned*?
  - What can we *tell*?

This talk...

- "Knowledge farming" on "SILAP" models
  - "Knowledge farming": see below
  - "SILAP": how IV&V selects which tasks to perform for a given project;
- Results:
  - SILAP v.1: an open source version of SILAP `http://unbox.org/wisp/trunk/silap`;
  - MB1:
    - 433 SLIAPed software elements from NASA
    - divided into different project types.
    - `http://unbox.org/wisp/trunk/silap/data/mb1.csv`
  - Experiments showing:
    - Selected IV&V tasks often the *same*, despite processing *different* projects;
    - Identification of the *really different* project types that lead to different IV&V tasks

Disclaimer etc

- Disclaimer
  - 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.
  - While the planning&scoping team at Fairmont is currently reviewing this material....
    - the views expressed here-in are the author.
    - They do NOT reflect official NASA policy
    - They do NOT reflect the views of NASA civil servants.

- Acknowledgments
  - The research described in this talk was carried out at West Virginia University under a contract with the National Aeronautics and Space Administration.
  - This work would not have been possible without the advise, assistance, and access to data offered by the NASA software Independent Verification & Validation planning & scoping team (at Fairmont WV).
  - Special thanks to (in alpha order): Markland Benson, Ken Costello, Ken McGill, Christina Moats, Melissa Northey.

Knowledge Farming

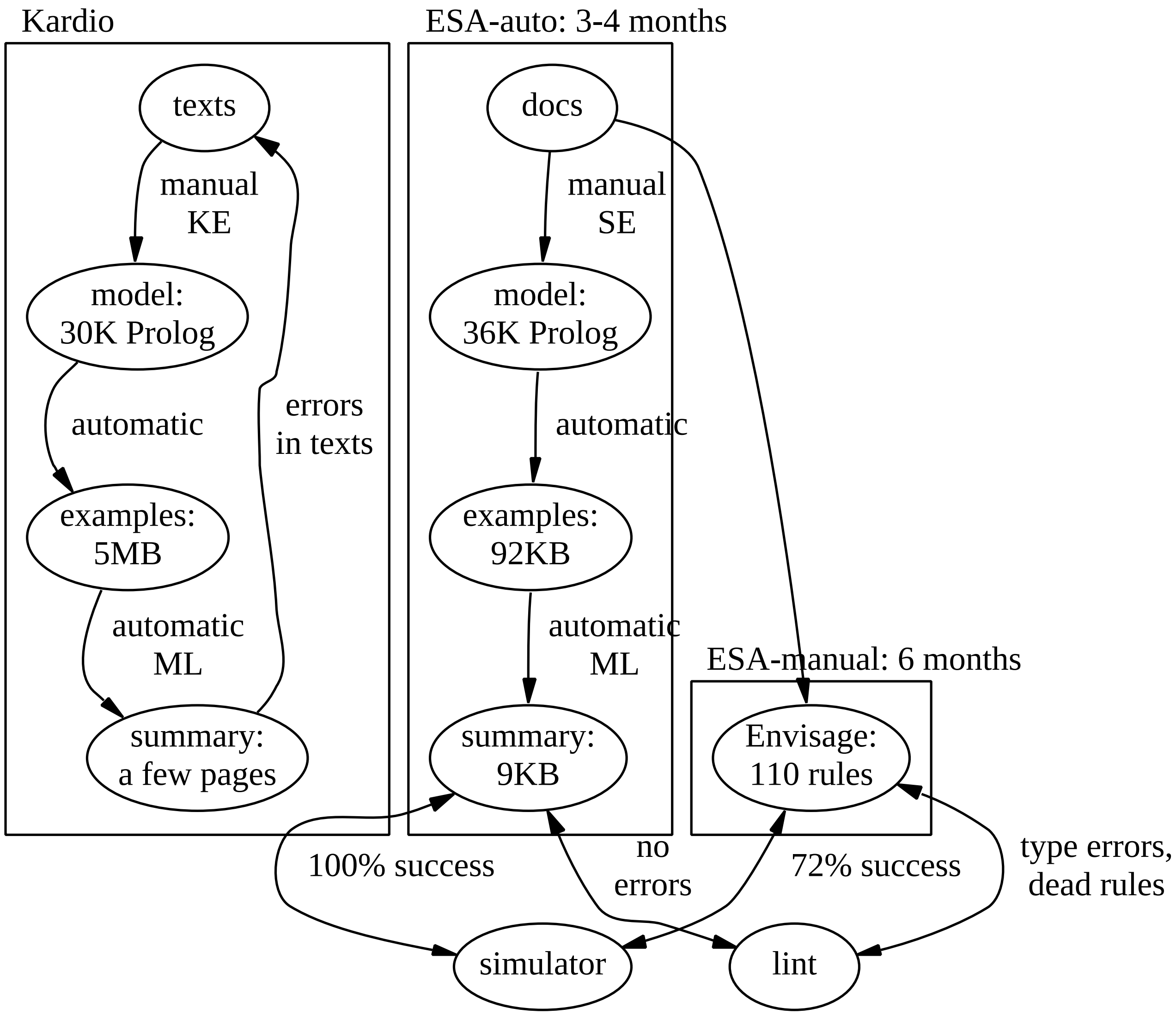
Data/Knowledge Farming

- Andrew Kusiak : J. Ops. research, 2005: "data farming"
  - DF generalizes DM (data mining)
  - DF = models + practices used to define most appropriate features for data collection/ transformation/ assessment
  - DF.effort greater than data mining effort
- DOD : "data farming"
  - Using a super-computer...
  - ... simulate the h#ck out a model
  - e.g. the Marine Corps' *Project Albert*
  - understand the landscape of potential simulated outcomes, enhance intuition, find surprises and outliers, and identify potential options
  - Heavily augmented with interactive visualization tools
  - ??? no automatic summarization methods (e.g. data miners)
- Me, ASE 2000: "Knowledge farming"
  - **Plant** the seeds; i.e. build a simulator that can reproduce domain conditions;
  - **Grow** the seedlings; i.e. Monte Carlo the simulator to generate data;
  - **Harvest** the crop; i.e. use data miners to find patters in the data;
  - Core problem: how to build human-readable succinct rules from gigabytes of data

The Case Against Knowledge Farming (KM)

- Q: Won't you just re-learn the original model?
  - A: Nope
  - Learned model = input data + summarization method;
  - Different summarization, different models;
  - KM + *relevancy tests* often prunes away variables that are noisy, low variance, under-sampled, not informative, etc etc
  - KM's harvest smaller than original.
  - Effects that are obscure in original are clear in summary
- Q: Aren't you just learning quirks in the model?
  - A_1 :If you have domain data, don't rely on a (possibly incorrect) model
  - A_2: What's the difference?
    - If the model is being used to make policy decisions, then we need to know the model, warts and all.
  - A_3: if your model has subtle quirks, how will you find them otherwise? (e.g. KARDIO)

Examples of knowledge farming (KARDIO: Bratko'89; ESA: Pearce'88)

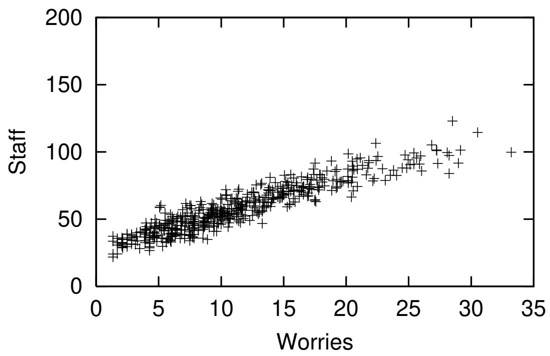


(More) Examples of knowledge farming (Menzies&Raffo: ASE’02; Menzies&Feather: RE’02)

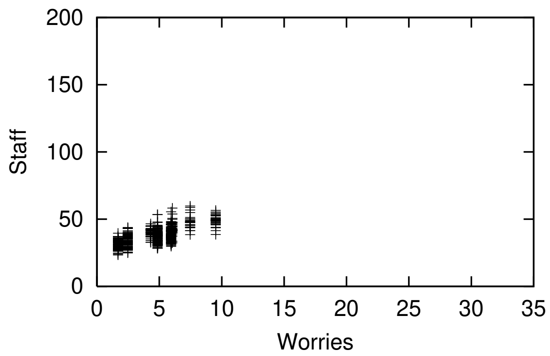
Introduction
Knowledge Farming
❖ Types
❖ Why not KM?
❖ Examples
❖ More examples
❖ Yet more examples
SILAP
mb1
What’s next?
Conclusion

Optimizing a combined staffing/“worries” model: staffing: the COCOMO-II effort model; “worries”: the Madachy software risk model.

before:



after:



Trade space studies at JPL: benefits: requirements coverage; costs: cost of risk mitigation strategies (each dot is yes-no to 99 decisions).

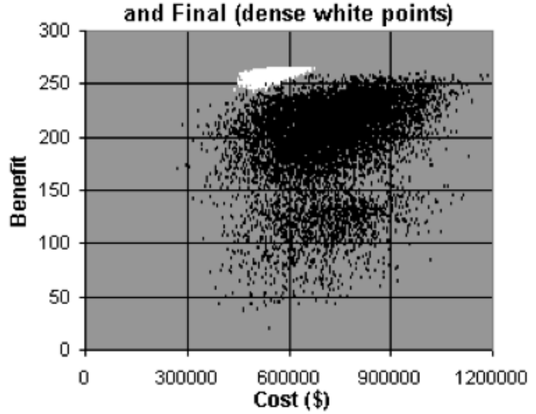


Figure 2. Initial (scattered black points) and Final (dense white points)

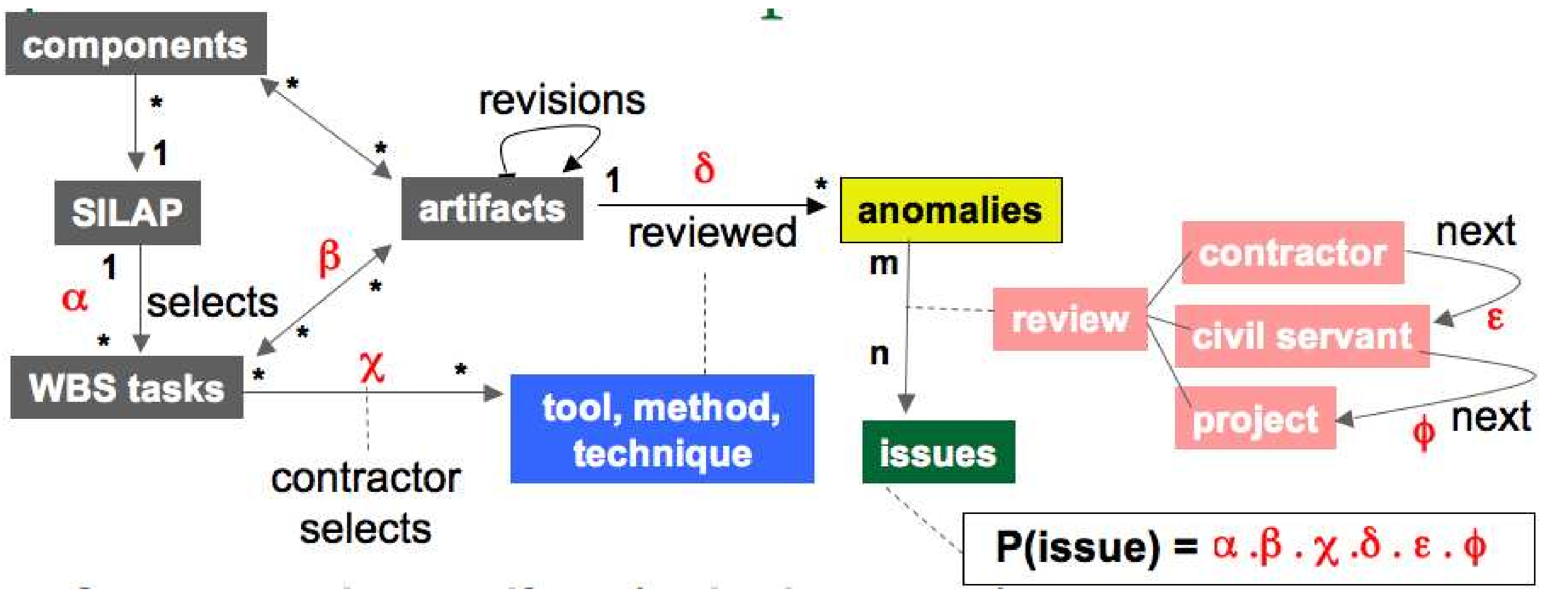
(Yet more) Examples

This talk: knowledge farming and SILAP

SILAP

The context of SILAP

- Components have artifacts
- SILAP selects WBS tasks
- Tasks are associated with artifacts;
- Artifacts generate anomalies;
- Filters reject bogus anomalies;
- Projects accept issues



SILAP: why yet another risk model?

- Software Integrity Level Assessment Process
- 16 criteria (scored 1,2,3,4,5)
  - Criteria used to calculate *errorpotential* (risk) and *consequence* (severity).
  - Which, in turn, select IV&V tasks from the IV&V work breakdown structure adapted from IEEE Std 1012 (V&V standard)
- In practice, takes two weeks (or more) full-time work to do a SILAP assessment.
  - Review the documents
  - Offer a score for the criteria
  - Write a detailed rationale for the score
- Understood locally (very important):
  - Previous risk models where developed elsewhere and, to a degree were “black box”; i.e. inexplicable, not defensible
  - SILAP, on the other hand, was built locally at Fairmont IV&V after *extensive* meetings between NASA civil servants.
- Civil servants report that SILAP has simplified and clarified their discussions with projects regarding what IV&V tasks are/are not to be performed.

SILAP makes IV&V business knowledge explicit, publicly accessible.

- This record lets outsiders (like me) review their work practices.
- How many other organizations can say they have done the same?

SILAP structure

variable = meaning	variable =meaning
AM3 = Artifact Maturity	FR3 =Use of Formal Reviews
AS2 = Asset Safety	HS2 =Human Safety
CL3 = CMM Level	PF2 =Performance
CO1 = Consequence	PR2 =Process
CX3 = Complexity	RA3 =Re-use Approach
DI3 = Degree of Innovation	RM3 =Use of Risk Management System
DO3 = Development Organization	SC2 =Software Characteristic
DT3 = Use of Defect Tracking System	SS3 =Size of System
DV2 = Development	UC3 =Use of CM
EP1 = Error Potential	US3 =Use of Standards
EX3 = Experience	

```
function CO1(   tmp) { // Consequence
    tmp=0.35*value("AS2") + 0.65 *value("PF2"); return round((value("HS2") > tmp) ? value("HS2") : tmp;) }
```

```
function EP1()  { // Error Potential
    return  round(0.579*DV2() + 0.249*PR2()  + 0.172*SC2()) }
```

```
function SC2() { // Software Characteristic
    return  0.547*value("CX3") + 0.351*value("DI3")    + 0.102*value("SS3") }

function DV2()  { // Development
    return  0.828*value("EX3") + 0.172*value("DO3") }

function PR2()  { // Performance
    return 0.226*value("RA3") + 0.242*value("AM3") + formality() }

function formality() {
    return 0.0955*value("US3") + 0.0962*value("UC3")  + 0.0764*value("CL3")
     +0.1119*value("FR3") + 0.0873*value("DT3") + 0.0647*value("RM3") }
```

SILAP selects tasks from the IV&V work breakdown structure

Introduction
Knowledge Farming
SILAP
❖ Context
❖ SILAP
❖ SILAP structure
❖ WBS
mb1
What's next?
Conclusion

wbs	factor	CO11	CO12	CO13	CO14	CO15	EP11	EP12	EP13	EP14	EP15
1.1	Management and Planning of IV&V	X	X	X	X	X	X	X	X	X	X
1.2	Issue and Risk Tracking		X	X	X	X		X	X	X	X
1.3	Final Report Generation		X	X	X	X		X	X	X	X
1.4	IV&V Tool Support		X	X	X	X		X	X	X	X
1.5	Management & Technical Review Support	X	X	X	X	X	X	X	X	X	X
1.6	Criticality Analysis	X	X	X	X	X	X	X	X	X	X
2.1	Reuse Analysis*			X	X	X					
2.2	Software Architecture Assessment			X	X	X					
2.3	System Requirements Review			X	X	X				X	X
2.4	Concept Document Evaluation				Z	Z					Z
2.5	SW/User Requirements Allocation Analysis				Z	Z					Z
2.6	Traceability Analysis				Z	Z					Z
3.1	Traceability Analysis - Requirements		X	X	X	X				X	X
3.2	Software Requirements Evaluation			X	X	X				X	X
3.3	Interface Analysis - Requirements				X	X			X	X	X
3.4	System Test Plan Analysis			X	X	X					
3.5	Acceptance Test Plan Analysis					X					
3.6	Timing and Sizing Analysis									Z	Z

{CO1, EP1} computed via SILAP
CO1 = "consequences of failure"
EP1 = "error potential"

X = "or";
* = for $RA3 > 1$;
Z = for human-rated flights

Work breakdown structure (cont.)

wbs	factor	CO11	CO12	CO13	CO14	CO15	EP11	EP12	EP13	EP14	EP15
4.1	Traceability Analysis – Design							X	X	X	X
4.2	Software Design Evaluation				X	X				X	X
4.3	Interface Analysis – Design					X				X	X
4.4	Software FQT Plan Analysis		X	X	X	X					
4.5	Software Integration Test Plan Analysis									X	X
4.6	Database Analysis								X	X	X
4.7	Component Test Plan Analysis										X
5.1	Traceability Analysis – Code					X		X	X	X	X
5.2	Source Code and Documentation Evaluation				X	X			X	X	X
5.3	Interface Analysis – Code				X	X			X	X	X
5.4	System Test Case Analysis				X	X					
5.5	Software FQT Case Analysis				X	X					
5.6	SW Integration Test Case Analysis										X
5.7	Acceptance Test Case Analysis					X					
5.8	SW Integration Test Procedure Analysis										X
5.9	SW Integration Test Results Analysis									X	X
5.1	Component Test Case Analysis										X
5.11	System Test Procedure Analysis					Z					
5.12	Software FQT Procedure Analysis					Z					
6.1	Traceability Analysis – Test		X	X	X	X					X
6.2	Regression Test Analysis									Z	Z
6.3	Simulation Analysis					Z					
6.4	System Test Results Analysis				X	X					
6.5	Software FQT Results Analysis				X	X					
7.1	Operating Procedure Evaluation					Z					
7.2	Anomaly Evaluation					Z					
7.3	Migration Assessment					Z					
7.4	Retirement Assessment					Z					

MB1: 433 SILAPed software elements

MB1: Components

433 SILAP-ed software elements from NASA (system, subsystem, CSCI, or CSC level); sanitized data, i.e. mission/center names replaced with *x1, x2,....* Divided into:

- All
- "Orbits"
  - gs1: ground system
  - es2: earth orbit
  - xf3: "transfer" (a mission that does not have an Earth-centric orbit),
  - go4: ground ops on different planet
- "profile": type of science
  - {es,hs,op,ss,su}
- "prime": who built the sub-system (which NASA center)
  - {p1,p2,p3,p4}
- Auto: learned by a clustering algorithm (EM) that ignores all the above distinctions
  - {cluster0, cluster1,cluster2,cluster3,cluster4}

Q: for the purposes of IV&V, which of the above matter at all?
A: apply the *principle of separation*

Principle of separation

- Different things get effected different ways.
  - $Out = f(In)$
  - *Weak separation:* All Ins don’t have the same Outs
  - *Strong separation:* Different Ins have different Outs
- A model that does not *separate* is *blunt*; i.e. is not sensitive to changes in the inputs.
- I.e. the SILAP divisions that matter are those that:
  - Are most different to the All set;
    - “Profile” selects for nearly the same tasks as “All”
  - Most select different WBS tasks. different tasks.
    - All the manual divisions (“orbits”, “prime”, “profile”) select for very similar tasks.
    - Only the automatic divisions separate from each other and from “All”
- Test for separation
  - For each division, summarize the distributions;
  - Run each distribution through SILAP; record selected WBS tasks;
  - Compare the selected WBS tasks.

SILAP input parameters: distributions
"All" and "orbits"

All: 433 records

HS2	3 =0.00 5=0.02 2=0.02 4=0.05 1=0.35 0=0.56
AS2	5 =0.07 2=0.07 4=0.10 3=0.10 1=0.67
PF2	2 =0.09 5=0.13 4=0.16 1=0.19 3=0.43
EX3	5 =0.05 4=0.14 2=0.17 3=0.21 1=0.44
DO3	0.5=0.01 3=0.02 1=0.08 5=0.11 2=0.33 4=0.45
US3	5 =0.04 3=0.21 2=0.24 1=0.50
UC3	3 =0.07 2=0.28 1=0.65
CL3	2 =0.00 5=0.04 4=0.20 3=0.76
FR3	3 =0.01 2=0.34 1=0.65
DT3	4 =0.01 3=0.31 2=0.34 1=0.35
RM3	1 =0.18 3=0.36 2=0.45
RA3	2 =0.07 3=0.11 4=0.12 5=0.12 1=0.58
AM3	3 =0.15 2=0.21 1=0.64
CX3	5 =0.04 2=0.11 4=0.12 1=0.31 3=0.41
DI3	4 =0.02 3=0.02 2=0.12 1=0.85
SS3	1 =0.06 4=0.07 2=0.30 3=0.57

How to read these tables

- rows sorted left-to-right as rarest-to-more-frequent;
- column one are SILAP variable names (see code)

e.g. the last line of the above table;

- Least common SS3 value = "1": occurs 6% of the time;
- Most common SS3 value = "3": occurs 57% of the time;
- "2" occurs 30% of the time;

Orbit = eo2 (265 records)

HS2	3 =0.01 5=0.01 2=0.02 4=0.08 1=0.41 0=0.48
AS2	5 =0.03 2=0.08 4=0.09 3=0.11 1=0.69
PF2	5 =0.09 2=0.10 4=0.19 1=0.25 3=0.37
EX3	5 =0.07 2=0.15 3=0.17 4=0.17 1=0.44
DO3	0.5=0.02 3=0.03 5=0.04 4=0.45 2=0.46
US3	5 =0.07 3=0.14 2=0.22 1=0.57
UC3	3 =0.03 2=0.16 1=0.81
CL3	2 =0.00 5=0.07 4=0.14 3=0.79
FR3	3 =0.00 2=0.09 1=0.91
DT3	4 =0.01 3=0.11 2=0.31 1=0.56
RM3	1 =0.16 2=0.30 3=0.54
RA3	3 =0.00 4=0.01 2=0.12 5=0.12 1=0.74
AM3	3 =0.19 2=0.20 1=0.61
CX3	5 =0.04 2=0.07 4=0.11 3=0.37 1=0.41
DI3	4 =0.01 3=0.01 2=0.08 1=0.91
SS3	1 =0.01 4=0.04 2=0.41 3=0.55

Orbit = xf3 (93 records)

HS2	2=0.02 1=0.10 0=0.88
AS2	5=0.03 2=0.07 3=0.09 4=0.14 1=0.67
PF2	2=0.08 1=0.10 5=0.11 4=0.15 3=0.57
EX3	4=0.12 2=0.14 3=0.24 1=0.50
DO3	1=0.18 5=0.37 4=0.45
US3	3=0.21 2=0.24 1=0.55
UC3	3=0.03 1=0.36 2=0.61
CL3	4=0.39 3=0.61
FR3	1=0.29 2=0.71
DT3	1=0.03 2=0.48 3=0.49
RM3	1=0.23 2=0.77
RA3	3=0.12 1=0.39 4=0.49
AM3	3=0.03 2=0.13 1=0.84
CX3	5=0.03 4=0.11 2=0.20 1=0.21 3=0.46
DI3	3=0.03 2=0.13 1=0.84
SS3	2=0.13 1=0.27 3=0.60

SILAP input parameters: distributions “orbits” (cont.)

Orbit = go4 (57 records)

HS2	0=0.43 1=0.57
AS2	2=0.04 4=0.04 3=0.11 5=0.27 1=0.55
PF2	4=0.09 2=0.09 1=0.11 5=0.29 3=0.43
EX3	5=0.02 4=0.04 3=0.09 2=0.41 1=0.45
DO3	5=0.05 1=0.30 4=0.64
US3	1=0.27 3=0.27 2=0.46
UC3	1=0.27 3=0.32 2=0.41
CL3	4=0.23 3=0.77
FR3	3=0.07 1=0.25 2=0.68
DT3	3=1.00
RM3	1=0.25 3=0.27 2=0.48
RA3	4=0.05 1=0.30 3=0.64
AM3	3=0.21 1=0.38 2=0.41
CX3	1=0.09 5=0.11 2=0.12 4=0.20 3=0.48
DI3	3=0.05 4=0.09 2=0.30 1=0.55
SS3	2=0.18 3=0.82

Orbit = gs1 (20 records)

HS2	4=0.05 2=0.05 5=0.26 0=0.63
AS2	5=0.16 4=0.21 1=0.63
PF2	4=0.05 2=0.05 1=0.11 5=0.26 3=0.53
CX3	4=0.11 1=0.16 2=0.32 3=0.42
EX3	3=1.00
DO3	2=1.00
US3	3=1.00
UC3	1=1.00
CL3	3=1.00
FR3	2=1.00
DT3	2=1.00
RM3	2=1.00
RA3	5=1.00
AM3	1=1.00
DI3	1=1.00
SS3	4=1.00

SILAP input parameters: distributions "prime"

Prime = p1 (182 records)

HS2	2 =0.01 1=0.05 0=0.94
AS2	5 =0.01 2=0.04 4=0.08 3=0.08 1=0.78
PF2	2 =0.08 5=0.10 1=0.15 4=0.16 3=0.50
EX3	4 =0.06 5=0.10 3=0.23 2=0.28 1=0.33
DO3	0.5=0.03 3=0.04 1=0.09 2=0.16 5=0.23 4=0.45
US3	5 =0.10 1=0.17 3=0.29 2=0.44
UC3	3 =0.04 2=0.42 1=0.54
CL3	5 =0.10 4=0.27 3=0.63
FR3	2 =0.49 1=0.51
DT3	4 =0.02 1=0.02 3=0.28 2=0.69
RM3	3 =0.02 1=0.28 2=0.70
RA3	3 =0.04 2=0.12 5=0.18 4=0.27 1=0.39
AM3	3 =0.04 2=0.36 1=0.60
CX3	5 =0.04 2=0.09 4=0.12 3=0.33 1=0.41
DI3	4 =0.01 3=0.01 2=0.17 1=0.81
SS3	1 =0.15 2=0.28 3=0.57

Prime = p3 (139 records)

HS2	5=0.01 3=0.01 2=0.04 4=0.14 1=0.78
AS2	5=0.05 3=0.12 2=0.13 4=0.14 1=0.56
PF2	5=0.07 2=0.09 4=0.21 3=0.31 1=0.32
EX3	3=0.01 4=0.33 1=0.65
DO3	4=0.33 2=0.67
US3	3=0.01 1=0.99
UC3	3=0.01 1=0.99
CL3	2=0.01 3=0.99
FR3	3=0.01 1=0.99
DT3	3=0.01 1=0.99
RM3	3=1.00
RA3	3=0.01 1=0.99
AM3	3=0.25 1=0.75
CX3	5=0.01 2=0.11 4=0.12 1=0.35 3=0.42
DI3	4=0.01 3=0.01 1=0.98
SS3	4=0.07 3=0.45 2=0.48

prime=p2 (96 records)

HS2	1=0.35 0=0.65
AS2	4=0.03 2=0.05 3=0.13 5=0.18 1=0.61
PF2	1=0.09 4=0.12 2=0.13 5=0.23 3=0.43
EX3	5=0.01 4=0.02 2=0.25 3=0.29 1=0.42
DO3	5=0.06 1=0.20 4=0.74
US3	3=0.20 2=0.27 1=0.53
UC3	3=0.23 1=0.31 2=0.46
CL3	4=0.39 3=0.61
FR3	3=0.05 2=0.40 1=0.55
DT3	2=0.02 1=0.12 3=0.86
RM3	3=0.17 1=0.29 2=0.54
RA3	5=0.01 4=0.03 2=0.12 3=0.41 1=0.43
AM3	2=0.25 3=0.25 1=0.49
CX3	1=0.08 5=0.11 2=0.12 4=0.16 3=0.54
DI3	4=0.05 3=0.05 2=0.20 1=0.69
SS3	2=0.14 3=0.86

Prime = p4 (20 records)

HS2	4=0.05 2=0.05 5=0.26 0=0.63
AS2	5=0.16 4=0.21 1=0.63
PF2	4=0.05 2=0.05 1=0.11 5=0.26 3=0.53
EX3	3=1.00
DO3	2=1.00
US3	3=1.00
UC3	1=1.00
CL3	3=1.00
FR3	2=1.00
DT3	2=1.00
RM3	2=1.00
RA3	5=1.00
AM3	1=1.00
CX3	4=0.11 1=0.16 2=0.32 3=0.42
DI3	1=1.00
SS3	4=1.00

SILAP input parameters: distributions "profile"

Profile = es (29 records)

HS2	0=1.00
AS2	2=0.04 4=0.04 3=0.21 1=0.71
PF2	4=0.04 1=0.07 5=0.18 2=0.18 3=0.54
EX3	3=0.14 2=0.39 1=0.46
DO3	4=1.00
US3	3=0.07 1=0.46 2=0.46
UC3	3=0.07 2=0.39 1=0.54
CL3	3=1.00
FR3	1=0.46 2=0.54
DT3	2=0.14 1=0.39 3=0.46
RM3	3=0.14 2=0.39 1=0.46
RA3	1=0.14 2=0.86
AM3	2=0.04 1=0.39 3=0.57
CX3	5=0.14 1=0.14 4=0.29 3=0.43
DI3	2=0.04 1=0.96
SS3	1=0.04 3=0.46 2=0.50

Profile = op (85 records)

HS2	2=0.02 1=0.44 0=0.54
AS2	2=0.04 4=0.07 3=0.08 5=0.19 1=0.62
PF2	2=0.08 1=0.10 4=0.14 5=0.23 3=0.45
EX3	5=0.01 3=0.07 4=0.15 2=0.33 1=0.43
DO3	5=0.08 1=0.43 4=0.49
US3	1=0.23 3=0.38 2=0.39
UC3	1=0.23 3=0.23 2=0.55
CL3	4=0.15 3=0.85
FR3	3=0.06 1=0.21 2=0.73
DT3	2=0.02 1=0.04 3=0.94
RM3	3=0.19 2=0.38 1=0.43
RA3	5=0.01 4=0.21 1=0.25 3=0.52
AM3	3=0.18 2=0.27 1=0.55
CX3	5=0.08 1=0.08 4=0.18 2=0.20 3=0.45
DI3	3=0.04 4=0.06 2=0.35 1=0.56
SS3	1=0.12 2=0.12 3=0.76

Profile = hs (158 records)

HS2	3=0.01 5=0.04 2=0.04 0=0.08 4=0.13 1=0.69
AS2	5=0.06 3=0.10 2=0.11 4=0.15 1=0.57
PF2	2=0.09 5=0.09 4=0.19 1=0.29 3=0.34
EX3	3=0.13 4=0.29 1=0.57
DO3	4=0.29 2=0.71
US3	3=0.13 1=0.87
UC3	3=0.01 1=0.99
CL3	2=0.01 3=0.99
FR3	3=0.01 2=0.12 1=0.87
DT3	3=0.01 2=0.12 1=0.87
RM3	2=0.12 3=0.88
RA3	3=0.01 5=0.12 1=0.87
AM3	3=0.22 1=0.78
CX3	5=0.01 4=0.11 2=0.13 1=0.32 3=0.42
DI3	4=0.01 3=0.01 1=0.98
SS3	4=0.18 3=0.39 2=0.42

Profile = ss (111 records)

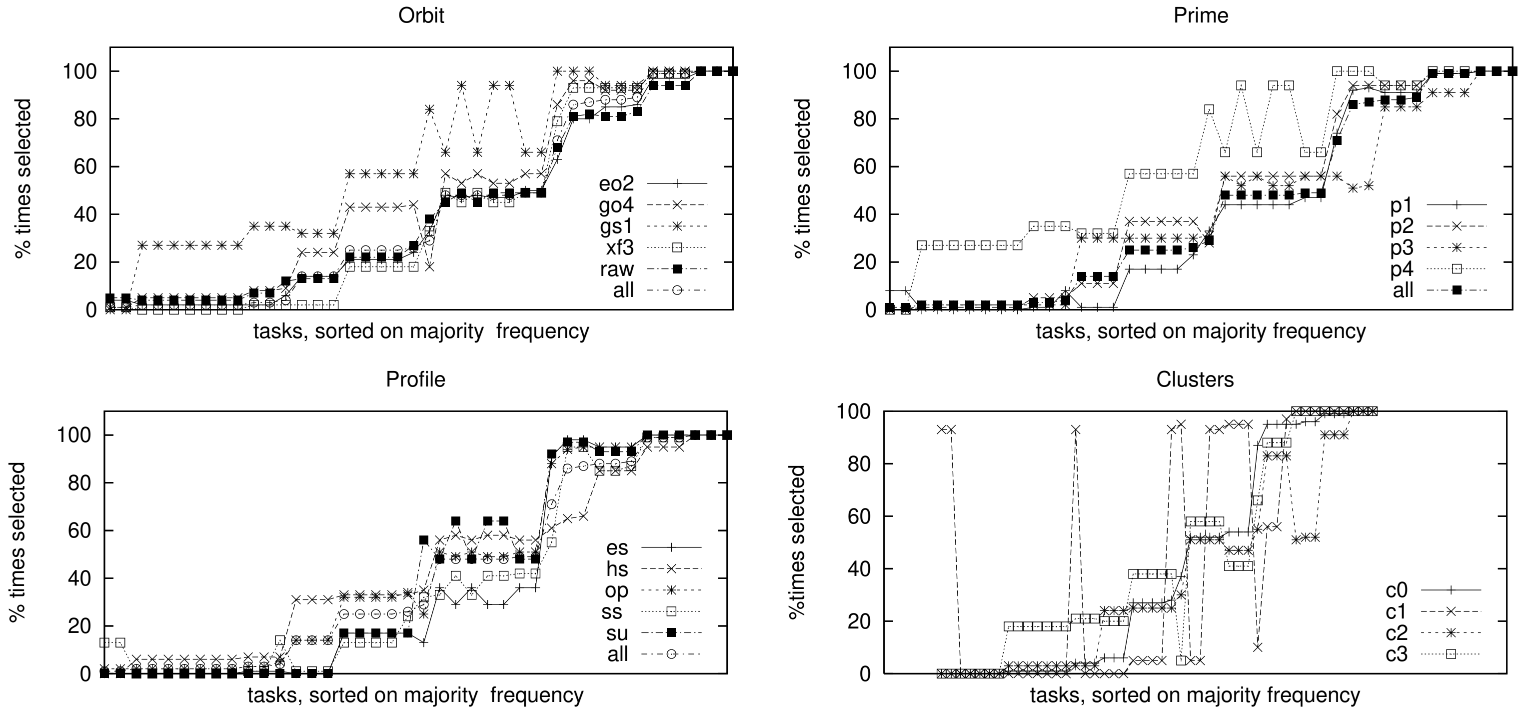
HS2	1=0.05 0=0.95
AS2	5=0.02 2=0.05 3=0.07 4=0.09 1=0.77
PF2	2=0.09 5=0.09 4=0.13 1=0.19 3=0.50
EX3	5=0.17 3=0.19 2=0.30 1=0.34
DO3	5=0.27 4=0.73
US3	2=0.14 5=0.17 3=0.25 1=0.44
UC3	3=0.03 2=0.47 1=0.50
CL3	5=0.17 4=0.33 3=0.50
FR3	2=0.38 1=0.62
DT3	3=0.22 2=0.78
RM3	1=0.17 2=0.83
RA3	3=0.03 4=0.27 1=0.70
AM3	2=0.28 1=0.72
CX3	5=0.02 4=0.06 2=0.07 3=0.29 1=0.55
DI3	3=0.03 2=0.18 1=0.79
SS3	1=0.14 2=0.34 3=0.53

SILAP input parameters: distributions "profile" (cont.)

Profile = su (55 records)

HS2	1=0.05 0=0.95
AS2	5=0.02 2=0.05 3=0.07 4=0.09 1=0.77
PF2	2=0.09 5=0.09 4=0.13 1=0.19 3=0.50
EX3	5=0.17 3=0.19 2=0.30 1=0.34
DO3	5=0.27 4=0.73
US3	2=0.14 5=0.17 3=0.25 1=0.44
UC3	3=0.03 2=0.47 1=0.50
CL3	5=0.17 4=0.33 3=0.50
FR3	2=0.38 1=0.62
DT3	3=0.22 2=0.78
RM3	1=0.17 2=0.83
RA3	3=0.03 4=0.27 1=0.70
AM3	2=0.28 1=0.72
CX3	5=0.02 4=0.06 2=0.07 3=0.29 1=0.55
DI3	3=0.03 2=0.18 1=0.79
SS3	1=0.14 2=0.34 3=0.53

500 times * {pick SILAP inputs from known distributions; compute (EP1,CO1); select relevant WBS tasks}



- “Profile” offers least separation;
- “Orbit” and “Prime” very similar
- “Orbits”, “prime”, and “profile” select for very similar tasks.
- “Clusters” offers the most separation

Sanity check: sampling naive?

- The proceeding experiment drew values from probability distributions of each SILAP variable, *ignoring* any correlations *between* variables.
- Q: are there large correlations?
  A: No

	HS2	AS2	PF2	EX3	DO3	US3	UC3	CL3	FR3	DT3	RM3	RA3	AM3	CX3	DI3	SS3
HS2		0.41	0.08	-0.06	-0.18	-0.26	-0.22	-0.29	-0.09	-0.42	0.47	-0.16	-0.05	0.07	0	0.13
AS2			0.44	-0.05	-0.06	-0.16	-0.05	-0.21	0.05	0.01	0.04	-0.01	-0.1	0.34	0.14	0.07
PF2				0.01	0.01	-0.02	0.07	-0.07	0.08	0.26	-0.15	0.16	-0.12	0.4	0.12	0.08
EX3					-0.19	0.45	0.21	0.47	-0.11	0.1	-0.19	0.05	0.41	-0.09	0.24	0.18
DO3						0.13	0.35	0.18	0.2	0.23	-0.13	-0.07	-0.05	-0.04	0.09	-0.23
US3							0.44	0.48	0.26	0.43	-0.47	0.29	0.03	-0.22	0.29	0.1
UC3								0.39	0.49	0.68	-0.23	0.08	0.14	0.13	0.34	0.01
CL3									-0.13	0.21	-0.31	0.03	0.24	-0.16	0.17	0.11
FR3										0.5	-0.12	0.53	-0.02	0.15	0.21	-0.03
DT3											-0.58	0.34	-0.12	0.24	0.3	0.09
RM3												-0.28	0.02	-0.04	-0.23	-0.12
RA3													0.02	0.15	0.06	0.2
AM3														0.01	0.11	-0.01
CX3															0.05	0.14
DI3																0.18
SS3																

- Besides, when we run the raw data for “all” through SILAP, it picks tasks very similar to the simulation for “all” .

What Separates the Clusters?

- "Tag" each record with its cluster identifier
- Using feature subset selection, learn which SILAP attributes are most important
  - 10-way, CFS, selected: HS2, EX3, US3, CL3, FR3, DT3, RM3
- Learn a decision procedure that identifies each clusters
- If SILAP performs differently for each cluster, then those clusters represent truly different project types.

```
DT3= use of defect tracking; CL3= CMM level; US3= use of standards;
EX3= experience;             HS2= human safety

DT3 <= 1: cluster2 (150.0/4.0)
DT3 > 1
|   CL3 <= 4
|   |   US3 <= 2: cluster0 (170.0)
|   |   US3 > 2
|   |   |   DT3 <= 2
|   |   |   |   EX3 <= 2: cluster3 (25.0)
|   |   |   |   EX3 > 2
|   |   |   |   |   HS2 <= 3: cluster0 (14.0)
|   |   |   |   |   HS2 > 3: cluster3 (6.0)
|   |   |   DT3 > 2: cluster0 (42.0)
|   CL3 > 4: cluster1 (19.0)
```

In a 10-way cross-val, accuracy=99.061% (!!).

"The variables in the tree are all about how much a project knows about itself and how much it is willing to share that knowledge with others."

What WBS tasks are Selected by the Clusters?

		W=231	X=21	Y=148	Z=32
id	task	c0	c1	c2	c3
1.1	Management and Planning of IV&V	100	100	100	100
1.2	Issue and Risk Tracking	99	100	91	100
1.3	Final Report Generation	99	100	91	100
1.4	IV&V Tool Support	99	100	91	100
1.5	Management & Technical Review Support	100	100	100	100
1.6	Criticality Analysis	100	100	100	100
2.1	Reuse Analysis*	87	10	55	66
2.2	Software Architecture Assessment	52	5	51	58
2.3	System Requirements Review	52	93	51	58
2.4	Concept Document Evaluation	6		24	20
2.5	SW/User Requirements Allocation Analysis	6		24	20
2.6	Traceability Analysis	6		24	20
3.1	Traceability Analysis - Requirements	95	97	83	88
3.2	Software Requirements Evaluation	52	93	51	58
3.3	Interface Analysis - Requirements	54	95	47	41
3.4	System Test Plan Analysis	52	5	51	58
3.5	Acceptance Test Plan Analysis	4		3	21
3.6	Timing and Sizing Analysis				
4.1	Traceability Analysis - Design	95	100	51	100
4.2	Software Design Evaluation	28	93	25	38
4.3	Interface Analysis - Design	4	93	3	21
4.4	Software FQT Plan Analysis	95	56	83	88
4.5	Software Integration Test Plan Analysis		93		
4.6	Database Analysis	37	95	30	5
4.7	Component Test Plan Analysis				

What WBS tasks are Selected by the Clusters? (more)

Introduction
Knowledge Farming
SILAP
mb1
❖ Components
❖ separation
❖ “All” and “Orbits”
❖ “Orbits” (cont.)
❖ “Prime”
❖ “Profile”
❖ “Profile” (cont.)
❖ Selections
❖ Separation
❖ Selected WBS
❖ Selected WBS (more)
What’s next?
Conclusion

		W=231	X=21	Y=148	Z=32
id	task	c0	c1	c2	c3
5.1	Traceability Analysis - Code	96	100	52	100
5.10	Component Test Case Analysis	96	100	52	100
5.11	System Test Procedure Analysis	1		3	18
5.12	Software FQT Procedure Analysis	1		3	18
5.2	Source Code and Documentation Evaluation	54	95	47	41
5.3	Interface Analysis - Code	54	95	47	41
5.4	System Test Case Analysis	27	5	25	38
5.5	Software FQT Case Analysis	27	5	25	38
5.6	SW Integration Test Case Analysis				
5.7	Acceptance Test Case Analysis	4		3	21
5.8	SW Integration Test Procedure Analysis				
5.9	SW Integration Test Results Analysis		93		
6.1	Traceability Analysis - Test	95	56	83	88
6.2	Regression Test Analysis				
6.3	Simulation Analysis	1		3	18
6.4	System Test Results Analysis	27	5	25	38
6.5	Software FQT Results Analysis	27	5	25	38
7.1	Operating Procedure Evaluation	1		3	18
7.2	Anomaly Evaluation	1		3	18
7.3	Migration Assessment	1		3	18
7.4	Retirement Assessment	1		3	18

But what *good* is any of this? Well...

- Check the above. Is it reasonable? If not- should SILAP be changed?
- Don’t make “blunt” business distinctions;
    - e.g. don’t plan IV&V around “mission type” but, rather, around the process maturity of the developers.
- Make business decisions that are sensitive to the kinds of current business; e.g.
- What are the *current* and *expected future* frequencies of {W,X,Y,Z}?
- Weight the above numbers by those frequencies.

What's next?

Supporting Process Improvement

The IV&V planning & scoping team are continually and actively reviewing and improving SILAP. Current activities include:

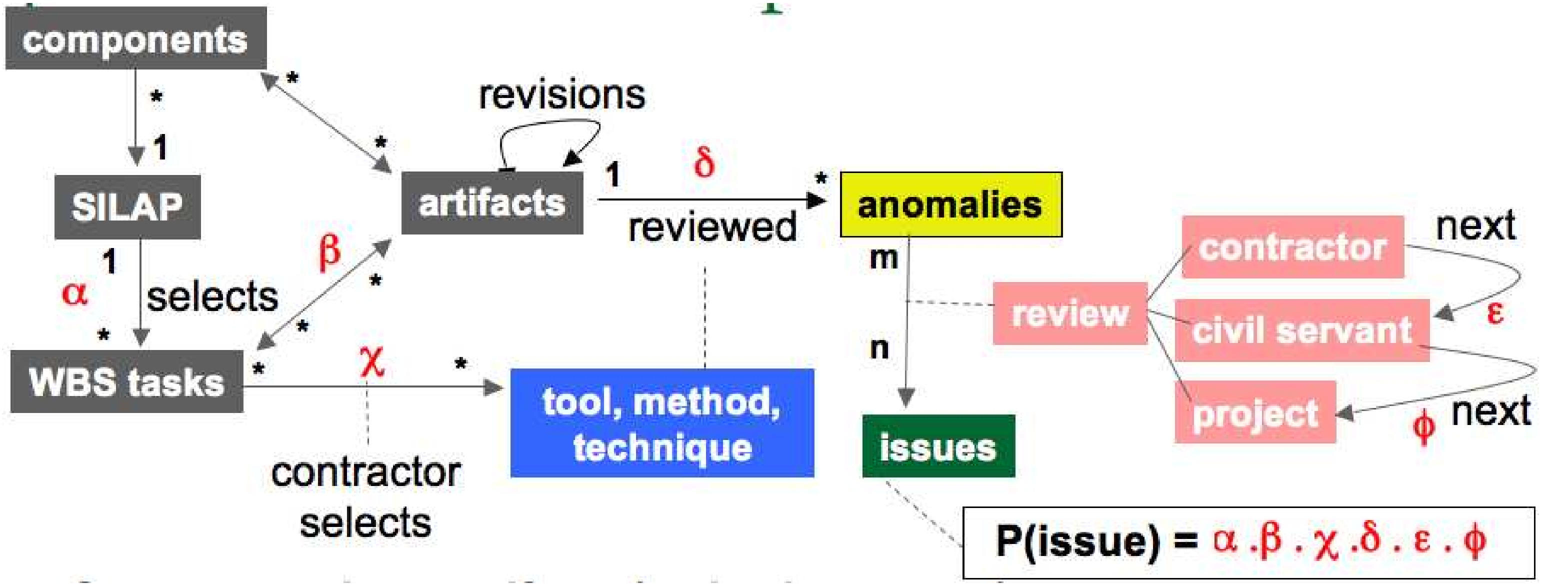
- Review/update SLP 9-1 (the WBS)
- Revising SILAP documentation
- identified areas for ?change:
  - right tasks selected by scores?
  - are factors the best selectors?
- Is the criteria sufficient/ correct?
  - Map factors directly to tasks?
  - Study planned vs actual to find a "best" or most common architecture break out?

  Minimum set of tasks needed to add IV&V value?

  - Defining sets of "common" tasks for specific types of functions?
- ?? break code analysis out into
  - tool execution only?
  - tool execution + review of results?
  - full-up code inspection?
- Etc etc

Can knowledge farming simplify, optimize any of these dialogues?

The Future



- This talk has been all about learning α.
- The rest of the alphabet awaits.

Conclusion

- If puzzled, then poke:
  - Model a little,
  - Simulate a little,
  - Summarize a little
- If certain, then check:
  - Model a little,
  - Simulate a little,
  - Summarize a little
- There are surprises hiding in your business.
- Use knowledge farming to find them.