UML-based Object-Oriented Metrics for Architecture Complexity Analysis

Peter In[1], SangEun Kim[2], Matthew Barry[3]

[1] Dept. of Computer Science, Texas A&M University, College Station, TX 77840 USA, Tel: (979) 458-1547, hohin@cs.tamu.edu
[2] Dept. of Computer Science, Texas A&M University, College Station, TX 77840 USA, Tel: (979) 845-5439, sangeunk@cs.tamu.edu
[3] United Space Alliance, 600 Gemini, Houston, TX 77058 USA, Tel: (281) 282-3960, Matthew.R.Barry@USAHQ.UnitedSpaceAlliance.com

Today's Agenda

- Motivations/Solution Approach
- Proposed Metrics
- Tool Support
- Experimental Results
- Conclusion

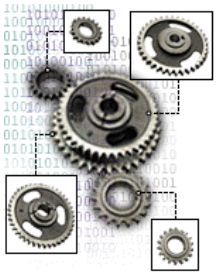
USA
United Space Alliance

Research Issues

1. What metrics will be helpful to a project manager early in the development lifecycle?
2. How can such metrics information be collected?
3. How can the generated metrics information be utilized?

Solution Approach

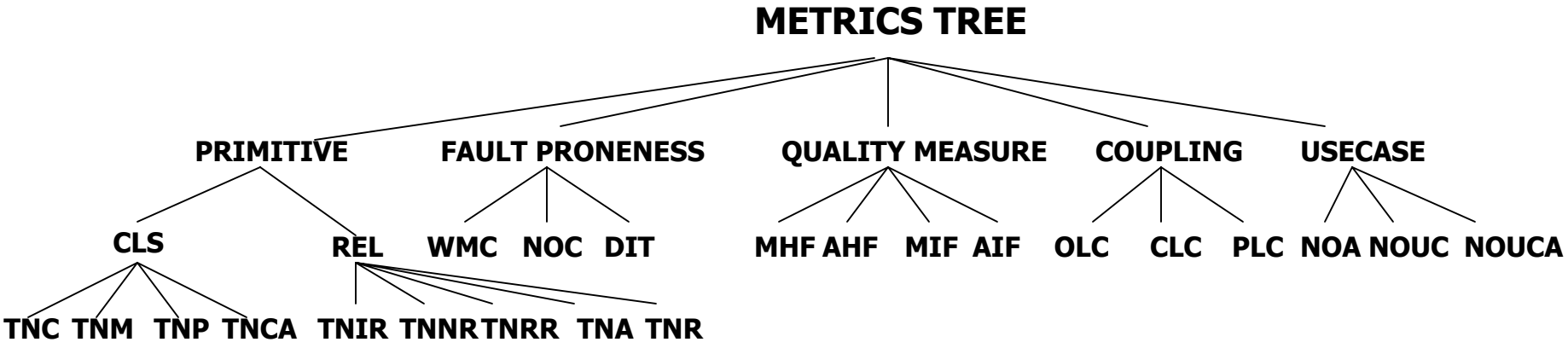
- UML-based Automatic OO Metrics Counter
- Input
  - UML diagrams: class, use case, component, deployment, state chart, activity, sequence, collaboration and package
- Output
  - Key metrics : model complexity, evolution, maturity, breakage, rework, etc.
  - Output format : Standard Output, XML, Excel, *et al*.



United Space Alliance

Proposed Metrics

- OO Concepts: Classes, Methods, Inheritance, Polymorphism
- Metrics Tree



United Space Alliance

Primitive Metrics

- To provide brief and basic complexity information
  - TNC (Total Number of Class) = $\sum_{i=1}^{n} tnc_i$
  - TNIR (Total Number of Inheritance Relationships) = $\sum_{i=1}^{n} tnir_i$
  - TNRR (Total Number of Realization Relationships) = $\sum_{i=1}^{n} tnrr_i$
  - TNUR (Total Number of Use Relationships) = $\sum_{i=1}^{n} tnur_i$
  - TNA (Total Number of Associations) = $\sum_{i=1}^{n} tna_i$
  - TNR (Total Number of Roles) = $\sum_{i=1}^{n} tnr_i$
  - TNO (Total Number of Operation) = $\sum_{i=1}^{n} tno_i$
  - TNP (Total Number of Parameters) = $\sum_{i=1}^{n} tnp_i$
  - TNCA (Total Number of Class Attributes) = $\sum_{i=1}^{n} tnca_i$

Fault-Proneness Metrics

- To predict class's fault-proneness
  - WMC(Weighted Method per Class) = $\sum_{i=1}^{n} c_i$

    where, c_i is the complexity of the methods
  - NOC(Number of Children per Class) = $\sum_{i=1}^{n} sc_i$

    where, sc_i is the number of immediate subclasses
  - DIT(Depth of Inheritance Tree) = max_leng

    where, max_leng is the maximum length from the root node to the leaf node

Quality Measure Metrics

- To provide quality measurements
  - MHF(Method Hiding Factor) = $\frac{\sum_{i=1}^{TC}\sum_{m=1}^{Md(c_i)}(1-V(M_{mi}))}{\sum_{i=1}^{TC}Md(C_i)}$

    where, $V(M_{mi}) = \frac{\sum_{j=1}^{TC} is\_visible(M_{mi}, C_j)}{TC-1}$

    $is\_visible(M_{mi}, C_j) = \begin{cases} 1 & iff \begin{cases} j \neq i \\ C_j \ may \ \ call \ \ M_{mi} \end{cases} \\ 0 & otherwise \end{cases}$

    TC = Total number of class

    Md = Total number of methods defined

    $V(M_{mi})$ = Visibility of the total classes from which the method M_{mi} is visible

  - MHF is a measure of the use of information hiding concept through methods

Quality Measure Metrics(Continued)

- AHF(Attribute Hiding Factor) = $\frac{\sum_{i=1}^{TC}\sum_{m=1}^{Ad(c_i)}(1-V(A_{mi}))}{\sum_{i=1}^{TC}Ad(C_i)}$

  where, $V(A_{mi}) = \frac{\sum_{j=1}^{TC} is\_visible(A_{mi}, C_j)}{TC-1}$

  $is\_visible(A_{mi}, C_j) = \begin{cases} 1 & iff \begin{cases} j \neq i \\ C_j \ may \ \ call \ \ A_{mi} \end{cases} \\ 0 & otherwise \end{cases}$

  TC = Total number of class

  Ad = Total number of attributes defined

  $V(A_{mi})$ = Visibility of the total classes from which the attribute A_{mi} is visible

- AHF is a measure of the use of information hiding concept through attributes

Quality Measure Metrics(Continued)

- MIF(Method Inheritance Factor) = $\frac{\sum_{i=1}^{TC} M_i(C_i)}{\sum_{i=1}^{TC} M_a(C_i)}$

  where, $M_a(C_i) = Md(C_i) + M_i(C_i)$ is total number of available methods(locally defined plus inherited)

- MIF is a measure of inheritance through methods

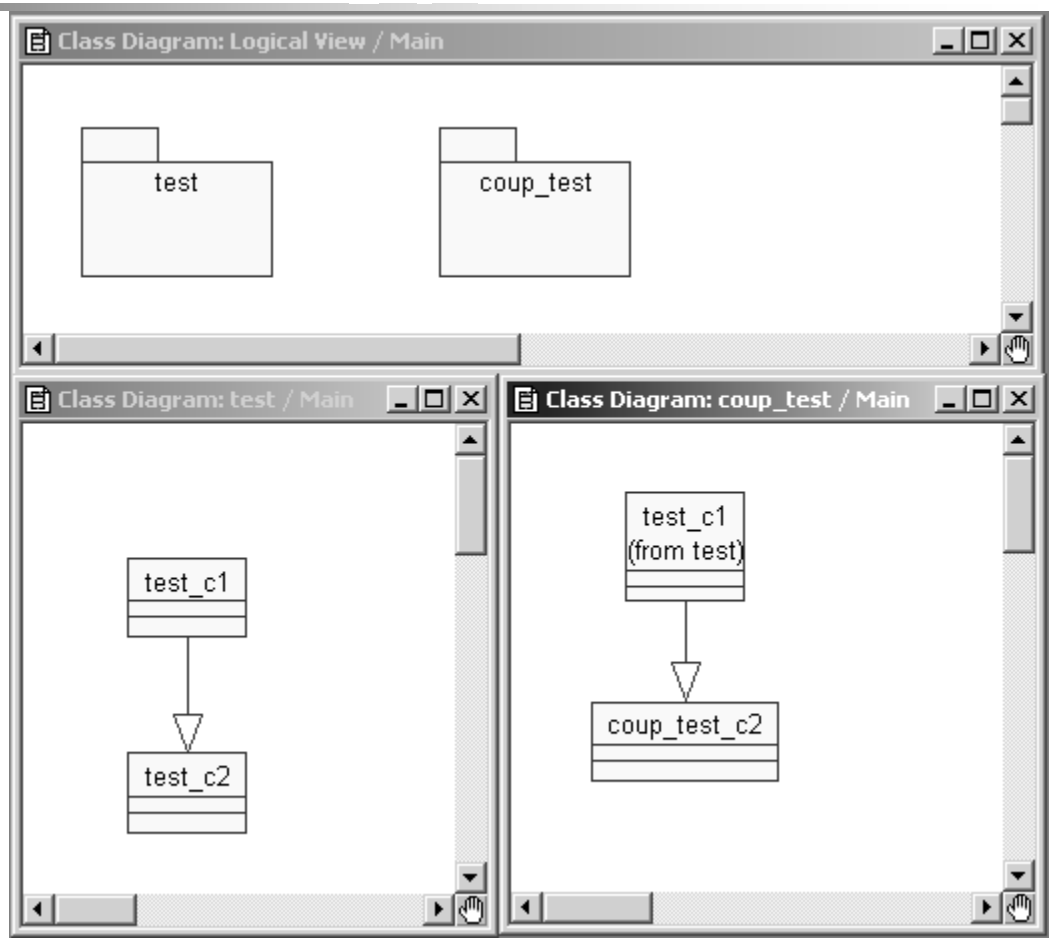
- AIF(Attribute Inheritance Factor) = $\frac{\sum_{i=1}^{TC} A_i(C_i)}{\sum_{i=1}^{TC} A_a(C_i)}$

  where, $A_a(C_i) = Ad(C_i) + A_i(C_i)$ is total number of available attributes(locally defined plus inherited)

- AIF is a measure of inheritance through attributes

Coupling Metrics

- To provide dependency between objects / classes and locality of data
  - PLC(Package Level Coupling)
  - CLC(Class Level Coupling)
  - OLC(Object Level Coupling)



<Example of PLC>

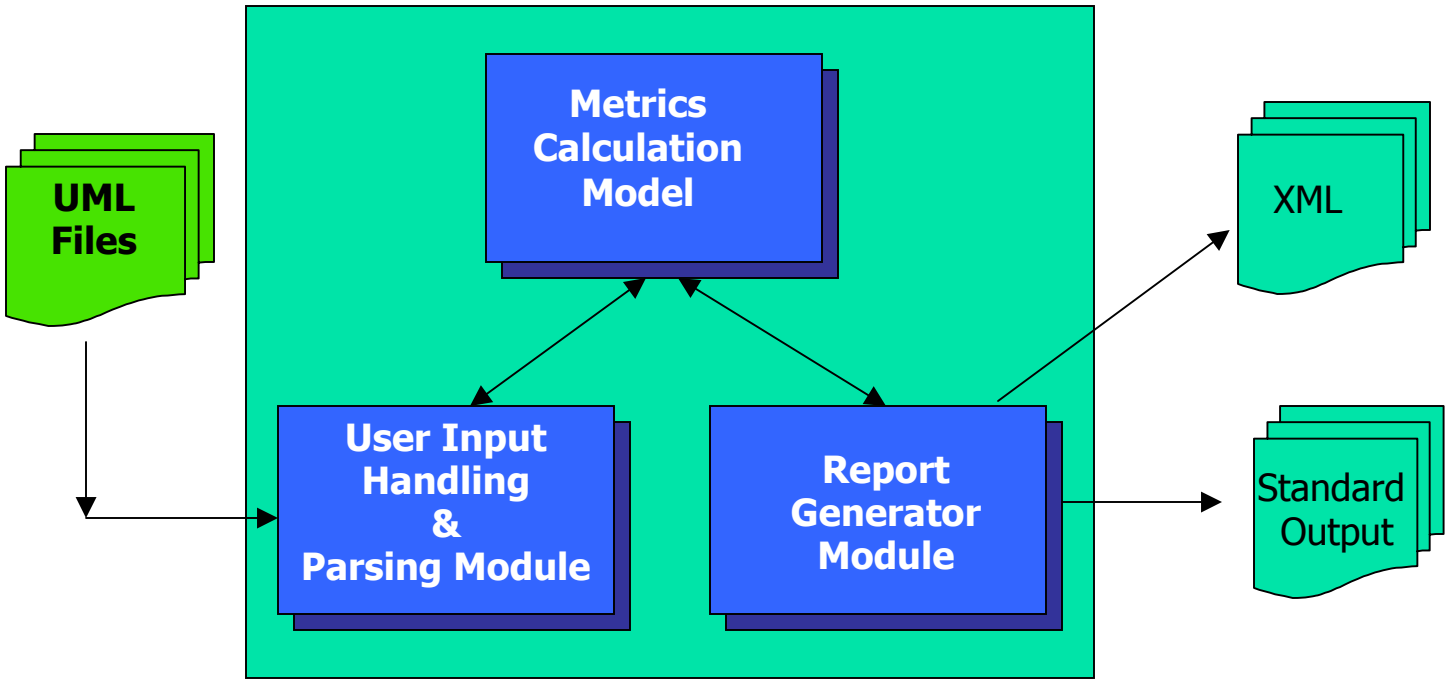
United Space Alliance

Use Case Metrics

- To provide dynamic complexity through the analysis of use case data
  - NOA(Number of Actor) = $\sum_{i=1}^{n} noa_i$
  - NOUC(Number of Use Cases) = $\sum_{i=1}^{n} nouc_i$
  - NOUCA(Use Cases per Actor) = $\sum_{i=1}^{n} nouca_i$

United Space Alliance

Tool Support

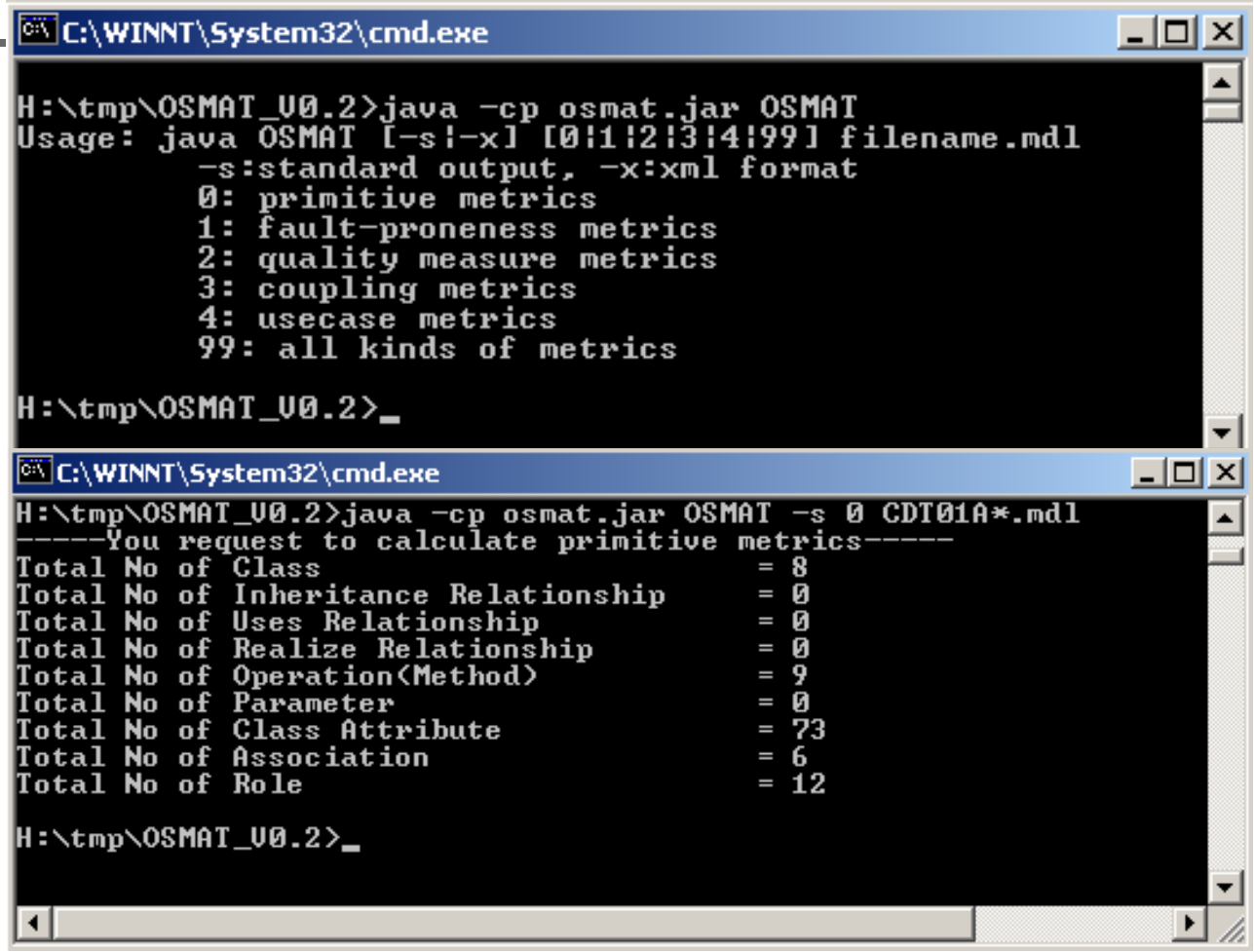


<Structure of the OSMAT software>

United Space Alliance

Screen shots

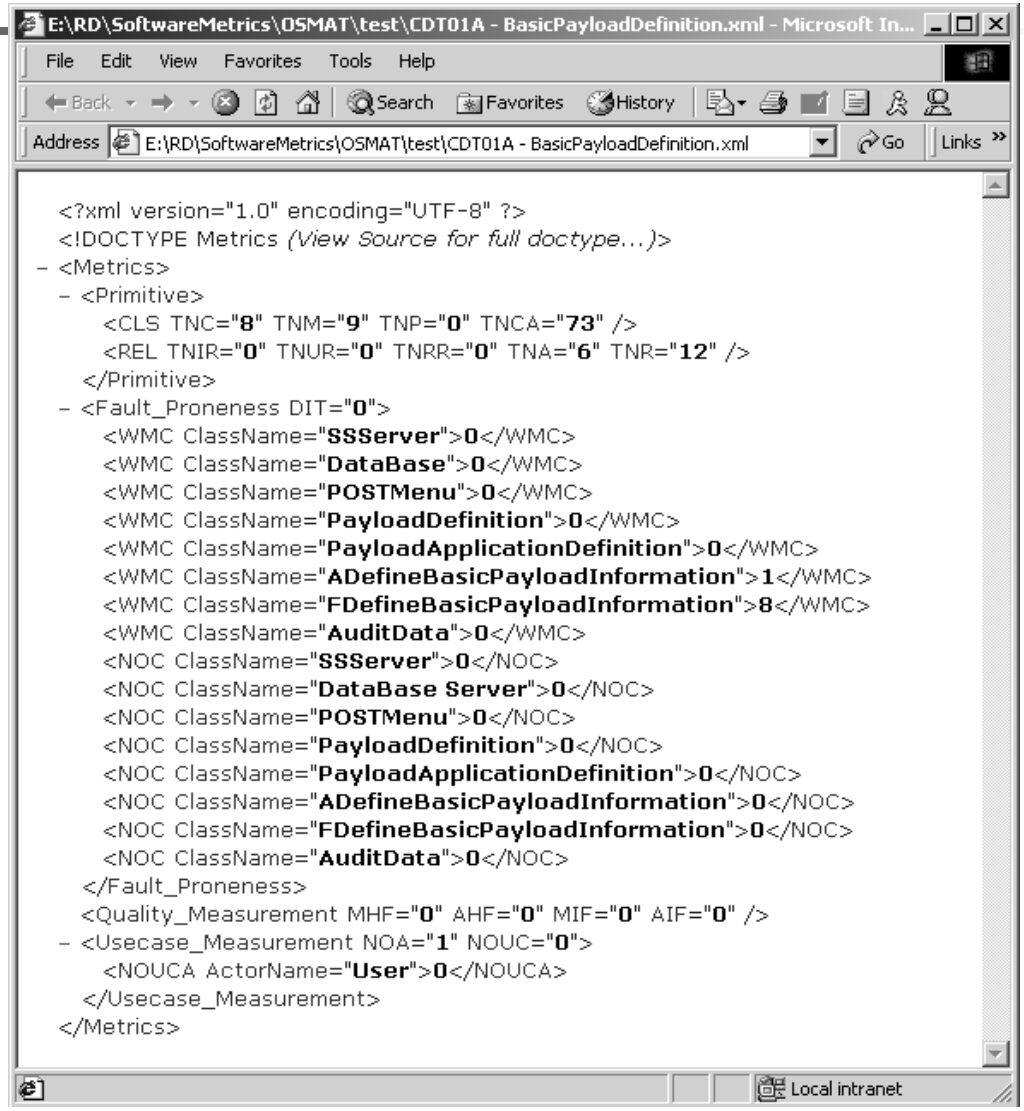
- Sample results of running the OSMAT



United Space Alliance

Screen shots(Continued)

- A sample XML output



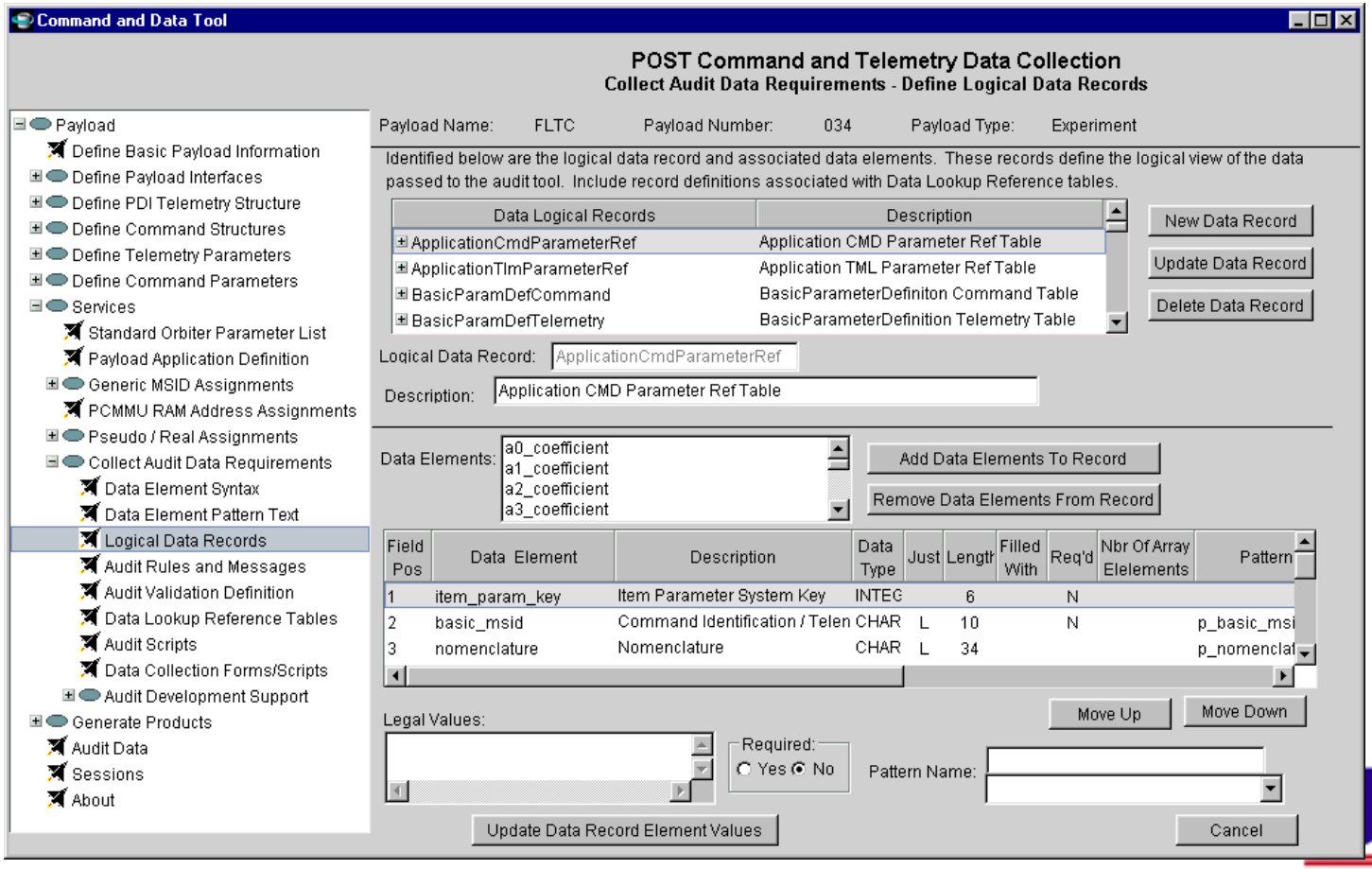
United Space Alliance

Experimental Results

- Feasibility test
  - CDT (Command and Data Tool) software for Payload Operations Support Team tool suite
    - Command and telemetry specification for Space Shuttle payloads and experiments
    - Java-language client-server application
    - Shipped to Space Shuttle payload customers
  - 30 UML models
    - Used RUP, UML and variety of tools
    - Collected back-end metrics
    - Wanted front-end metrics
      - Can we determine complexity earlier?
      - Use measures for project decision-making

Experimental Results(Continued)

- CDT Application Example



United Space Alliance

Experimental Results(Continued)

Results of running the OSMAT software

Metrics	File	File1	File2	File3	File4	File5	File6	File7	File8	File9	File10	File11	File12	File13	File14	File15
Basic	TNC	8	17	21	16	21	16	32	15	13	30	21	37	51	17	29
	TNIR	0	0	15	3	6	4	29	1	0	0	1	4	4	0	0
	TNUR	0	20	28	13	15	14	32	11	9	32	25	63	15	44	18
	TNRR	0	0	1	1	0	1	4	0	0	0	0	0	0	0	0
	TNO	9	147	35	11	9	24	65	10	19	63	41	92	49	35	113
	TNP	0	0	0	0	3	0	3	0	0	0	0	9	0	0	0
	TNCA	73	178	200	49	94	91	282	83	40	306	211	204	110	141	312
	TNA	6	7	0	1	5	0	2	2	0	2	11	18	27	6	103
	TNR	12	14	0	2	10	0	4	4	0	4	22	36	54	12	210
Fault-Proneness	DIT	0	0	1	1	0	1	0	1	0	0	0	1	1	0	0
Quality	MHF	-0.0317	-0.0557	-0.0214	0.0242	0.0722	-0.0250	-0.0159	0.0286	-0.0307	-0.0181	-0.0256	-0.0169	0.0004	-0.0339	-0.0269
	AHF	-0.1292	-0.0565	-0.0428	-0.0340	-0.0346	-0.0938	-0.0277	-0.0516	-0.3875	-0.0249	-0.0453	-0.0197	-0.0084	-0.0492	-0.0295
	MIF	0.5000	0.5000	0.5000	0.5000	0.4375	0.5000	0.4922	0.5000	0.5000	0.4960	0.5000	0.5000	0.5000	0.5000	0.5000
	AIF	0.5000	0.4986	0.4885	0.4494	0.4749	0.6111	0.4919	0.4713	0.8319	0.4496	0.5000	0.4701	0.4660	0.4739	0.4783
Others	NOA	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0
	NOUC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Metrics	File	File16	File17	File18	File19	File20	File21	File22	File23	File24	File25	File26	File27	File28	File29	File30
Basic	TNC	37	45	11	40	57	11	55	21	43	45	14	5	139	99	17
	TNIR	0	0	0	2	4	0	0	0	0	5	0	0	0	1	10
	TNUR	1	4	16	20	6	3	54	86	27	41	13	0	0	5	2
	TNRR	0	0	0	0	2	0	0	1	0	0	0	0	0	0	9
	TNO	118	616	12	63	7	21	61	73	99	107	12	26	581	0	44
	TNP	10	1	0	0	0	3	11	16	0	0	0	0	222	0	32
	TNCA	329	384	79	97	393	33	407	102	124	317	78	27	0	601	18
	TNA	96	15	2	19	50	14	24	30	0	32	0	0	0	67	5
	TNR	192	30	4	38	102	28	48	60	0	66	0	0	0	134	10
Fault-Proneness	DIT	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0
Quality	MHF	-0.0193	-0.0211	-0.0167	-0.0098	0.1276	-0.0524	-0.0021	-0.0363	-0.0137	-0.0134	0.0064	-0.2115	-0.0052	0.0000	-0.0256
	AHF	-0.0659	-0.0081	-0.0709	-0.0116	-0.0137	-0.0667	-0.0039	-0.0044	-0.0142	-0.0670	-0.0542	-0.0741	0.0000	-0.0041	-0.0069
	MIF	0.5000	0.5000	0.5000	0.5000	0.4615	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.4996	0.0000	0.4884
	AIF	0.7129	0.3204	0.4552	0.4611	0.4760	0.4923	0.2546	0.2214	0.4833	0.7552	0.4658	0.3077	0.0000	0.3606	0.5000
Others	NOA	1	1	0	0	1	1	1	0	1	1	0	0	0	0	5
	NOUC	0	0	0	0	0	0	0	0	1	0	0	0	0	0	2

Utilization of the OSMAT

- Cost Estimation Model
  - Generated by SAS with linear regression test

Model Equation											
TOT	=	3091.31	-	148.363	TNC	+	20.7095	TNUR	-	546.326	TNRR
	-	5.5201	TNO	+	17.0502	TNCA	-	2031.25	TNA	+	997.562	TNR
	+	982.287	DIT	-	17822.3	MHF	+	21811.3	AHF	+	41021.4	MIF
	-	3161.79	AIF	+	2613.69	NOA	+	21.7577	NOUC	+	77.9870	TNP
	+	59.6933	TNIR	-	0.6004	P_18	-	9.5140	P_19	-	0.0337	P_20
	+	13.0528	P_21	-	2063.36	P_23	+	1.6016	P_25	-	0.4766	P_26
	+	521.317	P_27	+	0.0072	P_28	-	3.1399	P_29	-	83662.2	P_30
	+	59665.8	P_31	-	80324.5	P_32						

Summary of Fit			
Mean of Response	781.6807	R-Square	1.0000
Root MSE	.	Adj R-Sq	.

Parameter Information	
Parameter	Variable
1	INTERCEPT
2	TNC
3	TNUR
4	TNRR
5	TNO
6	TNCA
7	TNA
8	TNR
9	DIT
10	MHF
11	AHF
12	MIF
13	AIF
14	NOA
15	NOUC
16	TNP
17	TNIR
18	TNP*TNP
19	TNIR*TNIR
20	TNCA*TNCA
21	TNA*TNA
22	DIT*DIT
23	NOA*NOA
24	NOUC*NOUC
25	TNC*TNC
26	TNUR*TNUR
27	TNRR*TNRR
28	TNO*TNO
29	TNR*TNR
30	MHF*MHF
31	AHF*AHF
32	MIF*MIF
33	AIF*AIF

Utilization of the OSMAT (Continued)

- Correlation between metrics by T-test

Metrics	Pearson Correlation		
	Negative	No relation	Positive
TNC	AIF (-0.506)	-	TNP (0.700)
TNIR	TNA (-0.172)	-	TNRR (0.577)
TNUR	NOA (-0.221)	-	AHF (0.218)
TNRR	TNCA (-0.116)	-	NOUC (0.770)
TNO	AIF (-0.474)	TNA, TNR	TNP (0.638)
TNP	AIF (-0.583)	-	TNC (0.700)
TNCA	MIF (-0.549)	-	TNC (0.398)
TNA	MIF (-0.338)	TNO	TNCA (0.613)
TNR	MIF (-0.334)	TNO	TNCA (0.612)
DIT	TNO (0.339)	-	MHF (0.339)
MHF	TOT (-0.186)	-	DIT (0.339)
AHF	AIF (-0.602)	-	TNC (0.366)
MIF	TNCA (-0.549)	-	TNUR (0.162)
AIF	TNP (-0.583)	-	DIT (0.227)
NOA	TNUR (-0.221)	-	TNRR (0.755)
NOUC	DIT (-0.165)	-	TNRR (0.770)
TOT	TNO (-0.245)	-	TNIR (0.132)

Summary

- What Metrics?
  - Object-Oriented Metrics to understand software complexity
  - Propose new metrics based on Use Case diagrams
- How to collect?
  - Automatic Measurement Approach: Ontology-based Software Metrics Analysis Tool
- How to utilize?
  - Effort estimation model
  - Statistical analyses: Regression, Cluster

Contact Information

- Peter In

Assistant Professor
Dept of Computer Science
Texas A&M University
College Station, TX 77843-3112

Email: hohin@cs.tamu.edu
Web: http://faculty.cs.tamu.edu/hohin
Voice: +1–979-458-1547

USA
United Space Alliance