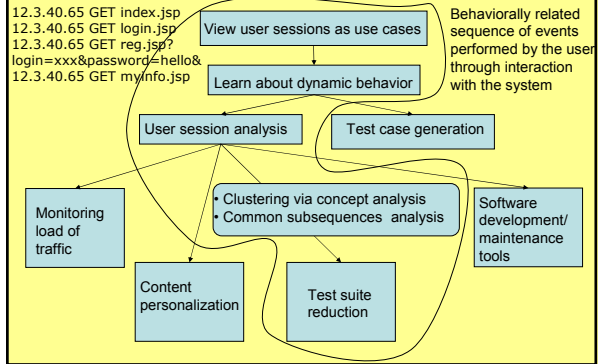


Towards Defining and Exploiting Similarities in Web Application Use Cases through User Session Analysis

Sreedevi Sampath, University of Delaware
 Amie Souter, Drexel University
 Lori Pollock, University of Delaware

Workshop on Dynamic Analysis (WODA), May 25, 2004
 co-located with
 International Conference on Software Engineering (ICSE 2004)

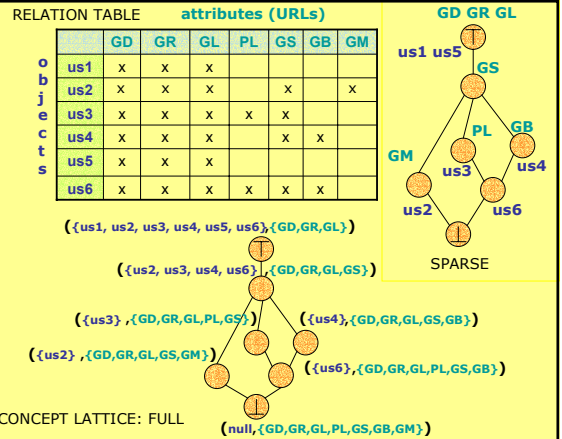
Motivation and Overview



Step 1

Clustering via Concept Analysis

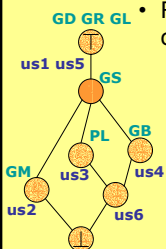
- Mathematical technique for clustering objects that have common discrete attributes
- Set of objects, **O**: user sessions, **us**
- Set of attributes, **A**: URLs, **u**
- Relation, **R**: **us** requests **u**
- Concept analysis identifies all **concepts** (O_i, A_j) for a given tuple (**O**, **A**, **R**)



Step 2

Heuristic for Test Suite Reduction

- Smallest set of user sessions
- Covers all the URLs
- Represents common URL subsequences of different use cases



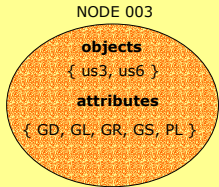
Identify *next-to-bottom* nodes
 Pick one user session from each of these *next-to-bottom* nodes
 Resulting reduced test suite: {us2, us6}

Hypothesis Motivating the Approach

- **Common Subsequences Hypothesis:**
 The set of user sessions clustered together into the same concept node will have a **high commonality in the subsequences of URLs** in their sessions

Finding Common Subsequences of URLs

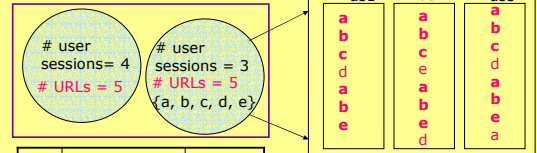
Subsequences of URLs are representative of **partial use cases** of the user sessions



us3	us6	Common Subsequences
GD	GD	[GD, GR, GL]
GR	GR	[PL, GS]
GL	GL	[GR, GL]
PL	GB	
GS	PL	
PL	GS	
GR	GR	
GL	GL	
PL	GB	
GS	PL	
	GS	

Metric for Common Subsequences Hypothesis

- attr-size[n] set: level of node in lattice
attr-size[5]: level 5



Sub seq size	Common subsequence	Percent attrs covered
1	a, b, c, d, e	100 %
2	ab, bc, be	80%
3	abc, abe	80%

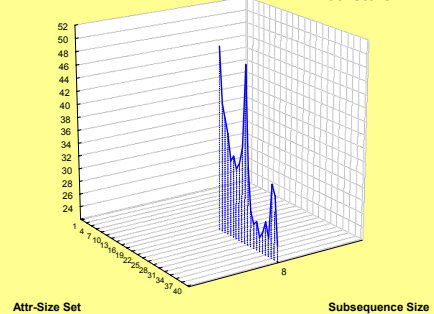
Metric
Percent of attributes covered by **common subsequences** of URLs of various sizes

Experiment: Applications Used

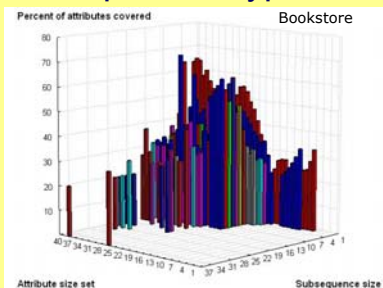
- Bookstore web application
 - 9,748 LOC, 385 methods, 11 classes
 - Front end: JSP, Backend: MySql
 - 123 user sessions
- uPortal application
 - 38,589 LOC, 4233 methods, 508 classes
 - Java, JSP, XML, J2EE
 - 2083 user sessions

Results for Common Subsequences Hypothesis

Percent of Attributes Covered Bookstore



Results for Common Subsequence Hypothesis



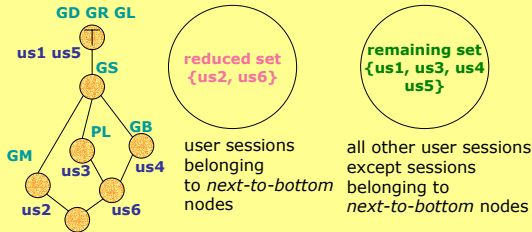
Result: subsequences of various sizes cover reasonable percent of attributes

Conclusions for Common Subsequences Hypothesis

- Between user sessions of a node **there exists commonality** in subsequences of URLs
- These common subsequences cover a **reasonable percent of URLs** (attributes) of the node
- Clustering based on single URLs
 - clusters similar use cases
 - can choose one object from each node

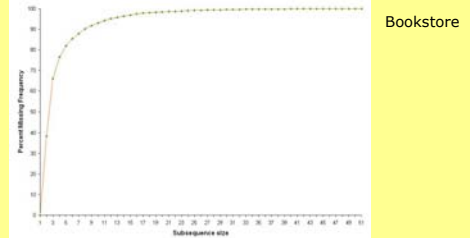
Next-to-bottom Coverage of Use Cases Hypothesis

In addition to covering all the URLs of the original test suite, the user sessions in *next-to-bottom* nodes execute a high percentage of the subsequences of URLs of the rest of the original test suite



Results for Next-to-bottom Coverage of Use Cases Hypothesis

Metric: loss of coverage of use cases in remaining set by the reduced set



Result: short sequences present but long sequences are missing

Conclusion for Next-to-bottom Coverage of Use Cases Hypothesis

- Long sequences absent but smaller sequences are present in **reduced set**
- **reduced set** contains more URLs hence may contain other URL sequences absent in **remaining set**
- Moderately supports picking **next-to-bottom** nodes for reduced test suite

Pros and Cons of Our Approach

- + Results from **common subsequences hypothesis** support using concept analysis for clustering user sessions
- + Experiments show little coverage loss (tech report) by reduced test suite
- Results from **next-to-bottom coverage of use cases hypothesis** indicate further work needed on heuristic

Future Work

- Explore additional heuristics
- Additional user session analysis
 - Useful for other software engineering tasks