

A Study of Architectural Information Foraging in Software Architecture Documents

Moon Ting Su, Ewan Tempero
University of Auckland
New Zealand

John Grundy
Swinburne University of Technology
Australia

John Hosking
Australian National University
Australia

Outline of Presentation

- Introduction
- Study Design
- Recruitment of Participants
- Results & Discussion
- Threats to Validity
- Conclusion

Introduction

- The difficulty of finding information in software documentation (Lethbridge, Singer et al., 2003).
- Also applicable to software Architecture Documents (ADs) (Koning and van Vliet, 2006).
- ADs may have inherent limitations, but the behaviour of those seeking information can also impact their usefulness.

Introduction

- Architectural information foraging in ADs.
- Information Foraging Theory (Pirolli, 2007)
 - assumes that humans are *informavores* (Miller, 1983), and so try to maximize the value of knowledge gained per unit cost of interaction (Pirolli, 2007).

Introduction

- Issues investigated:
 - types of forages (information diet)
 - commonly foraged information
 - foraging sequences
 - common sequences of foraging supporting better understanding
 - foraging styles
 - features of ADs that supported or hindered understanding.

Introduction

- Two groups of foragers: academics and industry professionals
 - Different perception of SA and reusable assets (Bosch, 1999).
 - Different emphasis on architectural information in ADs between the two groups?

Study Design

Study on Foraging



Explore AD for certain tasks

Participants



Industry Professionals



Academics

Tasks

Task 1

SA of the system?
Role: Software Architect (new)

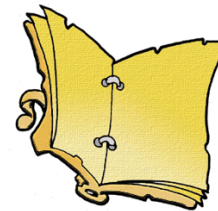
Task 2

How to change certain part of the system? Which parts are affected?
Role: Developer

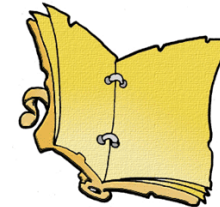
Task 3

How system was designed to achieve certain quality attribute?
Role: Maintainer

ADs



AD1: Digital Web Content Preservation



AD2: Storage Management Platform

Study Design

Data collected

Answer (bullet-point),
how was found

Highlighted
information

Task start &
stop time

Keywords

Questionnaire



Occupational
background,
experience (SA,
AD), foraging
styles, etc

Annotation widgets

The image shows two examples of an annotation widget interface. Each widget is a rectangular box with a blue border. At the top right of each widget is a text input field labeled "Suggested reading sequence:". Below this are two rows of radio button options. The first row is labeled "Importance to question:" and has five radio buttons numbered 1 to 5, followed by "Not Important." and "Not Sure.". The second row is labeled "Importance to overall understandability of SA:" and has the same five numbered radio buttons, followed by "Not Important." and "Not Sure.". Below these rows are two text input fields: "Tags:" and "Comment:". The first widget is titled "4.4.5 UC10 -- Scheduler" and contains the text: "The diagram below provides a high-level overview of the proposed design for managing the distributed nature of Harvesters as described UC8 and UC10, and the non-functional requirements of this system." The second widget is titled "2.1.1 Modularity/Plugability" and contains the text: "Modularity and Plugability requirements relate to the flexibility of the system and the ability to interchange components and external tools without additional coding effort. The requirement".

Recruitment of Participants

	Industry	Academic
Contacted	32	28
Responded	27	17
Excluded	11	3
Took Part	16	14
Dropped out	4	1
Completed	12	13
Excluded from analysis	0	1

Results & Discussion

- based on 4 industry and 4 academic participants performing Task 1 for the first AD.

Participants' Background & Perception

	Industry	Academic
Average years of SA experience in occupation	10.75	9.25
Average years of Designing, Changing SA	>	
Average years of Referring to, Reviewing SA		>
Experience in Reading, Reading & Making use of ADs in tasks, Writing ADs	=	=
Experience in Updating ADs	>	
Prior background with similar system	√	√
Ease of understanding AD language	√	√
Domain concepts were comprehensible	√	√

Architectural Information Forages

- Keywords provided

Industry Participants	Academic Participants
architecture goal, constraint, design decisions, framework, interface, overview, pattern, purpose, quality, software architecture and views.	modules, processes and system architecture.

- No repetition of keywords across different participants.
- Different pre-conceived ideas on what to look for with regards to the SA of a system prior to the exploration task.

Architectural Information Forages

- Commonly foraged information (most popular) based on answers & highlighted information

Industry	Academic
Main logical components	Main logical components, Components deployment, Process view
Quality requirements	
Purpose of the system	Use cases

Foraging Sequences

		Suggested Reading Sequence				
		Industry Participants		Academic Participants		
		E2	E3	E4	E8	E9
Order in the sequence	1	TOC (1)	TOC (1)	TOC (1) Quality Req. (3)	TOC (1)	TOC (1)
	2	Quality Req. (3) * External Dependency (4) *	Introduction (2)	Main logical components (6)	Use-case view (5)	Introduction (2)
	3	Main logical components (6)	Main logical components (6)	Arch design package - signi. use cases (9)	Main logical components (6)	Quality Req. (3) * External Dependency (4) *
	4	Size & performance (15)	Use-case view (5)	Distributed Nature (10)	Deployment view (12)	Size & performance (15)
	5	Resiliency & testing (16)	Logical deployment (13)	Process view (11) * Alternative deployment diagram (13) *	Introduction (2)	Resiliency & testing (16)

* Those in the same cell have the same order

Table III. First Five Sections in Suggested Reading Sequences

Foraging Sequences

- Foraging sequences starting with certain information were suggested to better support understanding of the described SA.
 - the overview of AD ('Table of Contents' and Introduction), main logical components, quality requirements, use cases and external dependencies
- Typically followed the written order of the information as dictated by the AD producers.

Foraging Styles

- Quite popular: referencing of table of contents, exploration based on titles and subtitles, skipping sections and forward-browsing long section
- Not popular: backtracking to previous section
- Main difference between the two groups
 - Referencing of ‘Table of Contents’ – majority of industry participants frequently did that.

Understanding Support & Hindrance

- Main support for understanding of the SA:
 - Industry participants: ‘views’
 - Academic participants: diagrams.
 - Combined group : diagrams, views and design decisions
- Main hindrance :
 - too much text with lack of diagrams.

Threats to Validity

- Non-probabilistic sampling techniques
 - the results not generalisable to the target population (Barbara and Shari Lawrence, 2002)
- Small number of participants
 - Participants had strong experience in SA (5.75 to 11.25 average years in various aspects)
 - Collectively good experiences in the production, and especially in the consumption of ADs
- Qualitative analysis (possible bias in data coding)

Threats to Validity

- Influence of AD on participants foraging
 - Use of second AD
- Instrumented AD may have affected the behavior of foragers
 - Participants were given the same instrumented AD
 - Focus on 'commonly' foraged information and foraging sequence

Conclusion

- There exists commonly foraged information and general foraging styles.
- Suggested foraging sequences typically followed the written order of the information in the ADs.
- Main support for understanding of the SA: diagrams, views and design decisions
- Main hindrance to understanding: too much text with lack of diagrams.
- Different emphasis on architectural information in ADs between industry and academics.

Acknowledgment

- Ministry of Higher Education, Malaysia
- PReSS, University of Auckland
- FRST Software Process and Product Improvement project
- All the participants who took part in this study



Q & A Session

Thank you.

References

- Ting, S.M., Hosking, J.G., Grundy, J.C., Tempero, E., A Study of Architectural Information Foraging in Software Architecture Documents, 10th Working IEEE/IFIP Conference on Software Architecture (WICSA) & 6th European Conference on Software Architecture (ECSA) - WICSA/ECSA 2012, August 20-24 2012, Helsinki, Finland, IEEE CS Pres.
- Ting, S.M., Hosking, J.G. and Grundy, J.C. Capturing architecture documentation navigation trails for content chunking and sharing , 2011 International Working Conference on Software Architecture (WICSA 2011), 21-24 June 2011, Boulder, Colorado.
- Ting, S.M., Hosking, J.G. and Grundy, J.C. KaitoroCap : a document navigation capture and visualisation tool, Demo Track of 2011 International Working Conference on Software Architecture (WICSA 2011), 21-24 June 2011, Boulder, Colorado.