

**Lero** THE IRISH SOFTWARE RESEARCH CENTRE

# Software Requirements Prioritisation - Research in Practice

University of Victoria V3

Kevin Ryan  
University of Limerick

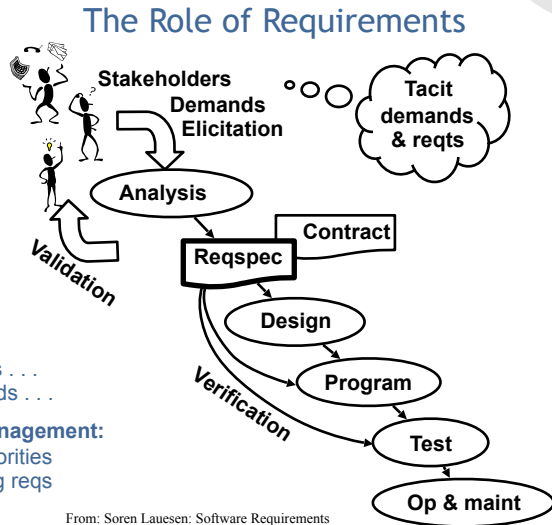
Lero 3/18/15 © Lero 2015

## Outline

- Part 1
  - What makes RE difficult ?
  - Why prioritise requirements?
  - Some Prioritising Techniques
  - The Analytic Hierarchy Process
  - An AHP Example

Lero 3/18/15 © Lero 2015

## The Role of Requirements



**Tracing:**  
Forwards . . .  
Backwards . . .

**Req. management:**  
Reqs Priorities  
Changing reqs

From: Soren Lauesen: Software Requirements  
© Pearson / Addison-Wesley 2002

Lero 3/18/15 © Lero 2015

## What makes RE difficult?

- All of the problems of Software Engineering
  - constant rapid change
  - overwhelming size and complexity etc. etc.
- Plus
- Capturing (just enough of) the Problem Domain
- Moving from the Informal to the Formal
- Deciding what is 'enough' - time, effort, detail
- Finding a suitable RE process
- Documenting the 'full' set of Requirements

• Deciding which ones to implement first - **Prioritise !!**

Lero 3/18/15 © Lero 2015

## Why Prioritise Requirements?

- We can't afford everything everyone wants
- Some requirements must wait until later versions (or forever!)
- Some requirements depend on, support or contradict others
- For long-lived software we expect to release many versions (possibly as a *software product line*)
- Stakeholders have different (implicit) priorities and different expertise
- Priorities are always set - if only by default or neglect
- But often in a way that is haphazard and without scientific basis

## Basics of Prioritisation

- We must decide what to implement
  - From all required (or unstated) requirements
  - Balance time-to-market v functions delivered
- Choose a level of Detail (function, feature, story)
  - How important is this?
  - How hard/expensive is it to implement?
  - How risky is it?
- Perform Triage
  - Some requirements are essential
  - Some are just too risky, difficult, expensive
  - That leaves the set we must choose from

## Some Prioritising Techniques

- Three level ranking: Must have; Want to have; Like to have
  - but... usually gives 85%, 10%, 5%
- Cumulative Voting - each stakeholder has a limited number of votes
  - but... gives no indication of relative importance
- \$100 to spend - distribute between requirements
  - but... some ought be cheaper than others in reality
- Benefit/Value/Risk estimated and combined (Wieggers)
  - but... assumes consistency over requirements & measures
- Real world problem is reconciling opposing claims

## The AHP

- Saaty's 'Analytic Hierarchy Process' (AHP) used in many fields for decision making
- Basic ideas taken from AHP
  - Enumerate the set to be prioritised
  - Compare them one pair at a time
  - Rate the relative priority of the pair on a ratio scale
  - Compute the cumulative priorities
  - Normalise them - so that total priority is 100%
- Produce a histogram of relative measures
- This is based on a complete enumeration of all pairwise comparisons ...
- In fact we can estimate consistency if more than a *spanning set* of pairs is compared

## Analytic Hierarchy Process

- Create nxn matrix (n requirements)
  - For each element (x,y) in the matrix enter:
    - 1 if x,y are of equal cost
    - 3 if x>y, 5 if x>>y, 7 if x>>>y, 9 if x>>>>y
    - put the reciprocal 1/3 etc in the y,x cell
- Average over the normalised columns
  - calculate the sum of each column
  - divide each cell by the sum of its column
  - calculate the sum of each row
  - divide each row by the number of rows
- The gives a cost for each requirement as a % of the total cost of all the requirements

## AHP example - estimating costs

	Req1	Req2	Req3	Req4
Req1	1	1/3	2	4
Req2	3	1	5	3
Req3	1/2	1/5	1	1/3
Req4	1/4	1/3	3	1

Normalise columns

Req1 - 26% of the cost  
Req2 - 50% of the cost  
Req3 - 9% of the cost  
Req4 - 16% of the cost

Result

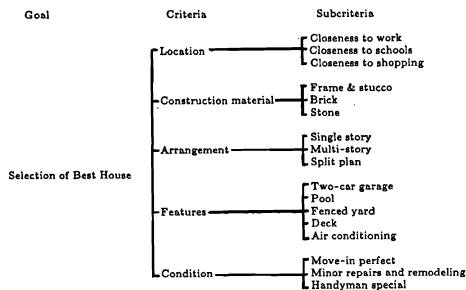
	Req1	Req2	Req3	Req4
Req1	0.21	0.18	0.18	0.48
Req2	0.63	0.54	0.45	0.36
Req3	0.11	0.11	0.09	0.04
Req4	0.05	0.18	0.27	0.12

Sum the rows

sum	sum/4
1.05	0.26
1.98	0.50
0.34	0.09
0.62	0.16

## AHP Applied to House Design

### Decision Hierarchy

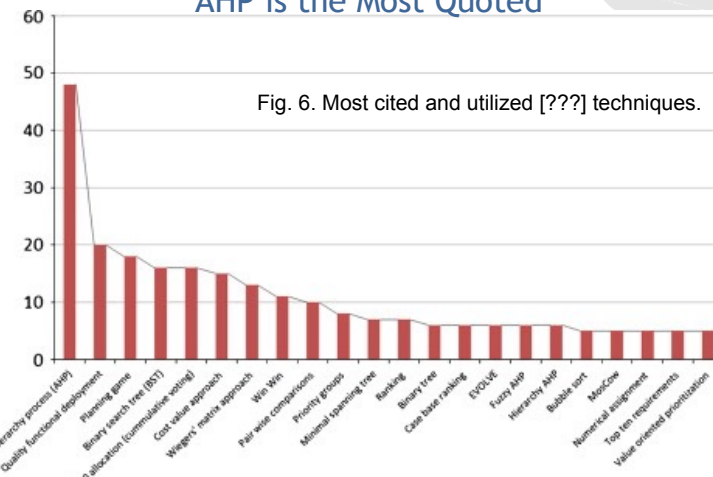


### Pairwise Comparison Scale (Saaty)

Value	Meaning
1	Equal importance
3	Moderate importance of one over another
5	Strong (essential) importance of one over another
7	Very strong (demonstrated) importance of one over another
9	Extreme importance of one over another
2, 4, 6, 8	Intermediate values
Reciprocals	Inverse comparisons

An AHP framework for prioritizing custom requirements in QFD: An industrialized housing application  
R. L. ARMACOST, P. J. COMPANION, M. A. MULLENS & W. W. SWART (1994) IIE Transactions 26:4, 72-79

## AHP is the Most Quoted



A systematic literature review of software requirements prioritization research Philip Achimugu, Ali Selamat, Roliana Ibrahim, Mohd Naz'ri Mahrin Information and Software Technology, Volume 56, Issue 6, 2014, 568-585

# Questions?

## Outline

- Part 2
  - The Cost-Value Approach
  - Industrial case study
  - Tool Support
  - Spin-off Company & Industrial Uptake
  - Publications & Impact on the field of RE
  - Summary of Lessons Learned

## Origins of the Cost-Value Approach

- Joachim Karlsson (U of Linköping & Ericsson) began a Masters on Requirements Engineering 1995
- He looked at techniques from other fields and found the AHP - previous lecture
- He combined two AHP histograms to produce a novel and very useful diagram
- He decided that in SE projects the most important metrics related to requirements are the **cost** of including each feature and its **value** to the customer
- But it is **more important** to support a productive discussion than to attempt to provide a deterministic process

## The Cost-Value Approach

- Consider separately
  - implementation cost and
  - stakeholder value of each Requirement
- Perform AHP comparison on each aspect
- Plot Cost ratings against Value estimates to give a scatter diagram
- High, medium and low value requirements can be easily seen
- Subset of potential requirements will usually deliver a high percentage of the possible value
- Always provokes a very worthwhile discussion !

## Cost and Value Histograms

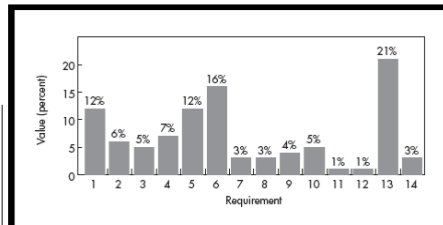


Figure 2. The value distribution of

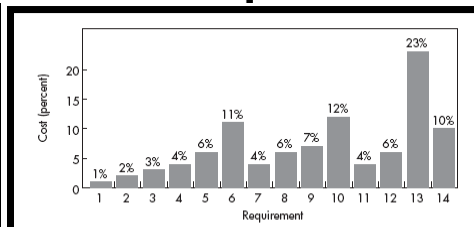


Figure 3. Estimated cost of requirements implementation in the RAN project. Requirements 6, 10, 13, and 14 constitute 56 percent of the total implementation costs.

Karlsson & Ryan "A Cost-Value Approach for Prioritizing Requirements". IEEE Software Vol. 14 No. 5. 1997

## The Cost-Value Diagram

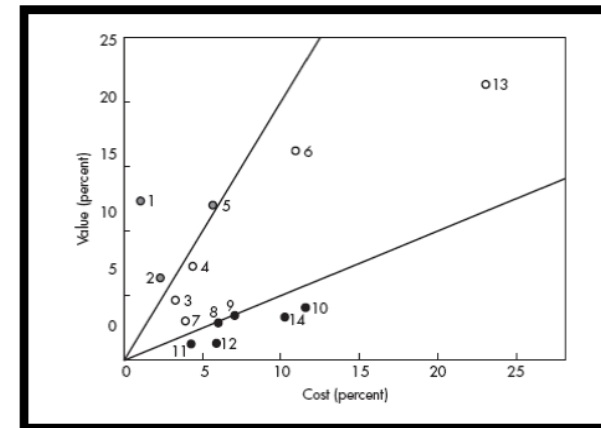


Figure 4. Cost-value diagram for the RAN project requirements. By not implementing the requirements that contribute little to stakeholder satisfaction, such as 10, 11, and 12, you can significantly reduce the cost and duration of development.

## There are Limitations ...

- Comparisons get tedious
  - $n*(n-1)$  needed to compare all  $n$  requirements
- Comparison Explosion at lower levels of any substantial system - (Full AHP structures decisions into a Hierarchy)
- What about linked requirements?
- How to deal with real (€) cost estimates?
- What about the rest of requirements management over the longer term?
- Next steps
  - Tool Support (prototype built by S Olsson)
  - Incorporate into commercial toolset (FocalPoint AB founded by J Karlsson)

## Prototype Tool Support

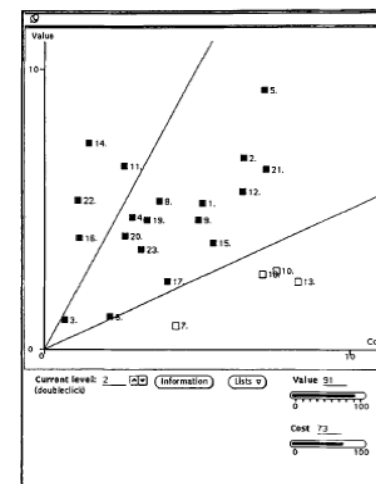


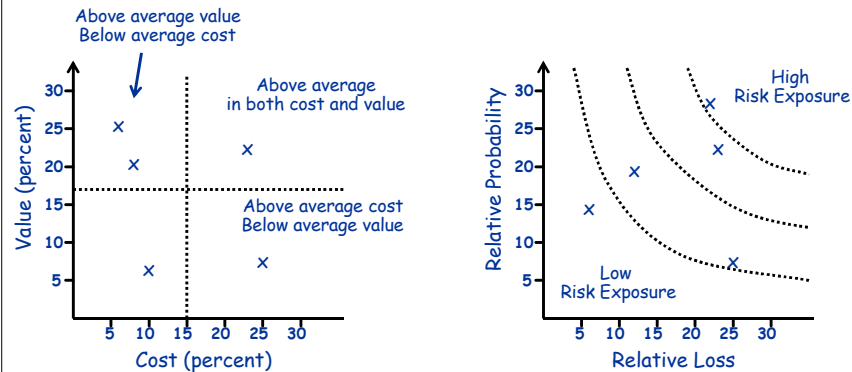
Fig. 3. Cost-value diagram for the 23 candidate requirements in the evaluation study.

### Tool provides

1. Stopping Rules to reduce no. of comparisons
2. Tracking of conflicts & dependencies
3. Measure of consistency
4. Measure of Cost and Value for selection
5. Repository of Requirements

Improved Practical Support for Software Requirements Prioritising Karlsson, Olsson, Ryan; Reqt Eng Journal 1997

## Alternative Diagrams



## Spin-off Company & Industrial Uptake

- Focal Point ab set up by J Karlsson in Linköping 1998
- Initially worked with Ericsson on software release planning
- Soon found non-software market - e.g. Whirlpool planning feature sets for washing machines
- Employment grew to 15 based on increased turnover
- In 2002 company relocated to California to target tool-vendor market
- 2005 Focalpoint bought by Telelogic - a leading provider of software development tools
- In 2007 Telelogic in turn bought by IBM
- Focalpoint technology incorporated into IBM toolsets where they are still being sold.

## Publications & Impact

“A Cost-Value Approach for Prioritizing Requirements”. IEEE Software Vol.14 No.5. 1997 [with J Karlsson]

*This paper has over 670 citations on Google Scholar (Jan 2016) and was the seminal paper for the field of Software Requirements Prioritisation. [see for example [http://en.wikipedia.org/wiki/Requirement\\_prioritization](http://en.wikipedia.org/wiki/Requirement_prioritization)]*

“Improved Practical Support for Large-scale Requirements Prioritizing” - Requirements Engineering Journal Vol.2 No.1, 1997, [with J Karlsson and S Olsson]

*This paper has over 110 citations on Google Scholar (Jan 2016)*

“Supporting the selection of software requirements” 8th International Workshop on Software Specification and Design, 1996 [with J Karlsson]

*This paper has 48 citations on Google Scholar (Jan 2016)*

See also

Joachim Karlsson, Claes Wohlin, Björn Regnell: “An evaluation of methods for prioritizing software requirements”. Information & Software Technology 39(14-15): (1998) (382 citations)

## Growth in the RE Research Field

### Paper publications distribution per year (1996-2013)

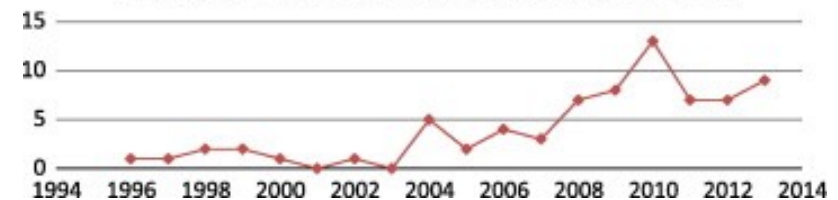


Fig. 4. Number of papers by year of publication.

**A systematic literature review of software requirements prioritization research** Philip Achimugu, Ali Selamat, Roliana Ibrahim, Mohd Naz'ri Mahrin. Information and Software Technology, Volume 56, Issue 6, 2014, 568-585

## Video ...

<http://download.boulder.ibm.com/ibmdl/pub/software/dw/demos/rfocalpoint/FocalPoint.html>

## Closing Remarks

- Labour-intensive methods of setting priorities may be worthwhile in highly critical software
- But most software is produced using **Agile Methods**
- For agile the requirements (called stories) are held in a 'backlog' - in priority order
- How are they prioritised? Is it 'scientific'?
  
- Bigger question: What's the future of Requirements Engineering in a world gone Agile?
  
- Take home message: An idea does not have to be complex or totally original for it to be useful and lucrative ...

## Questions?