

Requirements Elicitation & Specification

Lecture 4, DAT230, Requirements Engineering
Robert Feldt, 2011-09-08

Recap

- SWEBOOK gives overview of SE field
 - Good for newcomers and if you want to refresh
 - At master level: Good idea to directly to original sources; less need for “textbook” interpretations
- Basic RE terminology in SWEBOOK KA number 1
- Bespoke vs Market-Driven Software Development
- Stakeholder Identification
- Stakeholder analysis: influence & affected, expectations & interests

What is Req Elicitation?

What is Req Elicitation?

“The art of determining the needs of stakeholders”

What is Req Elicitation?

“The art of determining the needs of stakeholders”

“The process of discovering the requirements for a system by communication with stakeholders and through the observation of them in their domain”

Other sources of info?

- Stakeholders are key but also DOMAIN knowledge
- Problem/application domain
 - What is the problem? Who can explain it?
 - Process descriptions? Mission statements?
- History
 - Previous & current systems/solutions
 - Documentation, Old reqs & designs

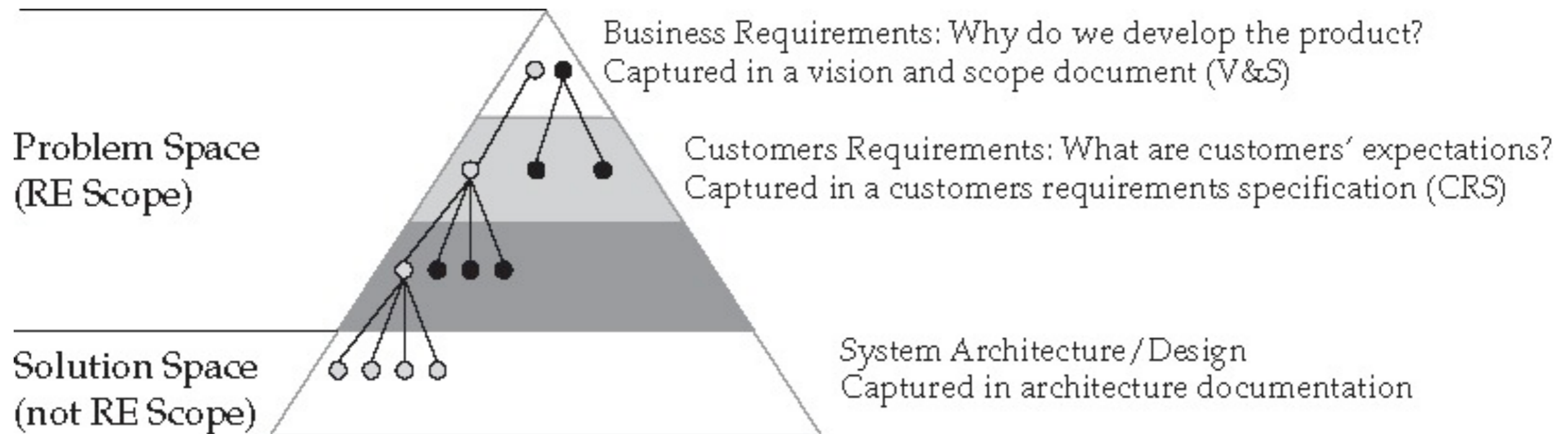
Other sources of info?

- Competitors
 - Is/are there a (partial) solution(s) out there?
- Environment
 - Other systems?
 - Processes to be supported? Processes that influence?
 - Organizational descriptions?

Information to elicit

- Domain description (operating environment)
- Business goals ... Technical goals
- System boundary (“fit into operational environment?”)
- Constraints
- Vocabulary
- Reqs
 - Title, description
 - Rationale, Source, Importance, Benefit, etc...

Differing abstraction levels



Differing abstraction levels

This is an example of two requirements specified on different levels of abstraction and at different levels of detail (i.e. more information is given in the case of Req. 2).

Requirement 1:

TITLE: "Support standardized formats"

DESC: "The system should support standardized formats"

Requirement 2:

ID: "X-11B"

TITLE: "Save output to XML"

DESC: "A user should be able to save output to a file in xml format in order for the data to be exported to the ERP system. Requirement O-7C needs to be implemented before this requirement."

SOURCE: "Kevin Incognito"

Requirements Abstract Model (RAM)

Organizational Strategies

Product Strategies

RAM - Abstraction Levels

Product Level (goal)

Feature Level (features)

Function Level (functions/actions)

Component Level (details- consists of)

Different elicited reqs

- **Discovered:** Stakeholder knows req - REng notes it
- **Created:** REng creates based on own knowledge or only little stakeholder info
- **Extracted:** REng uses method to find it
- **Captured:** When verbalized or acknowledged by stakeholder

General rules for elicitation

- Genuinely care about your stakeholders' problems
- Focus on stakeholder not on you "looking good"
- Be human - admit weaknesses, become vulnerable, show humor
- Listen - eye contact, don't glaze over
- Expect changes
- Maintain a glossary - many req problems from simple misunderstandings/miscommunication

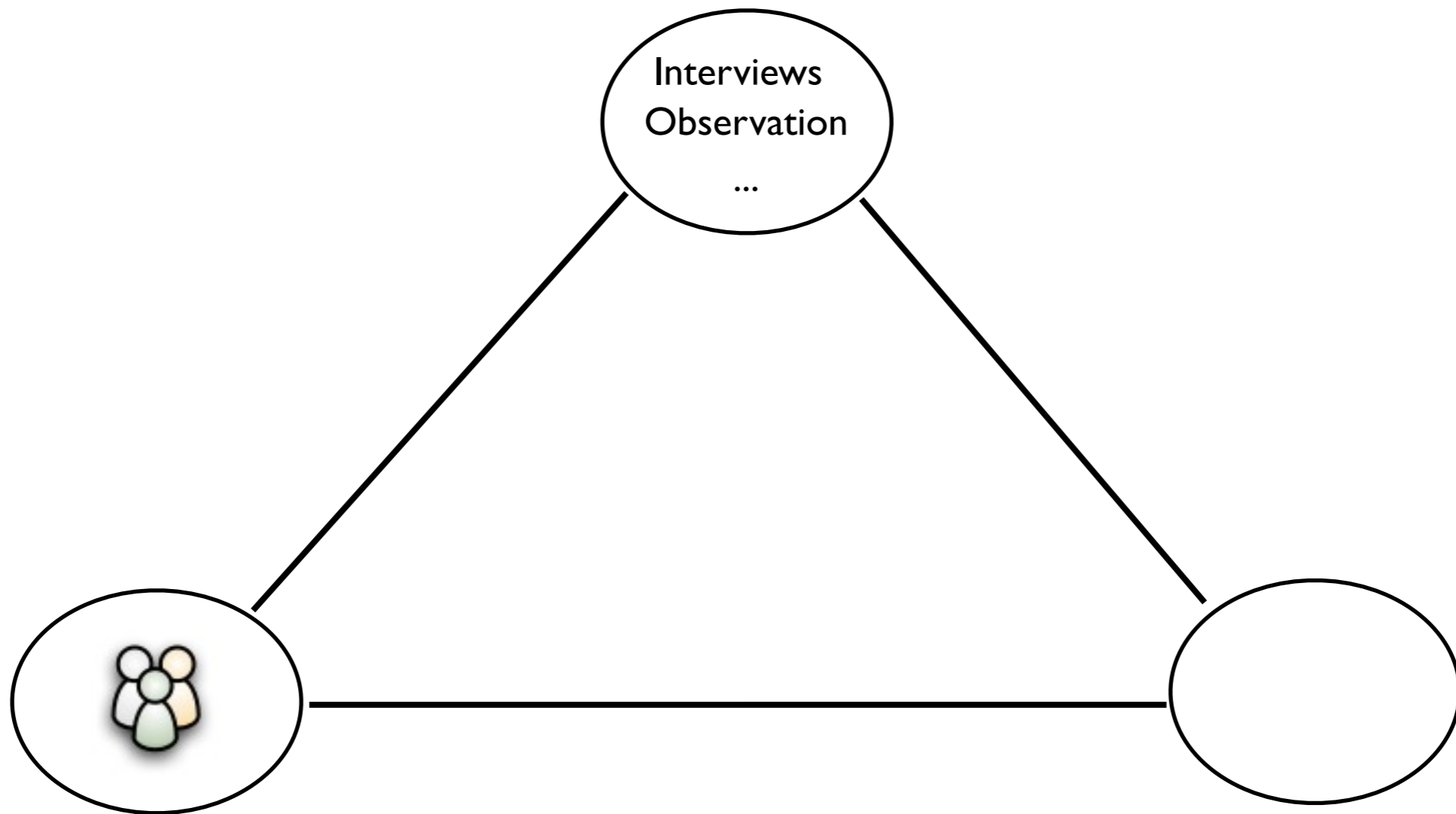
Triangulation

Use multiple things
so that they partly say (and thus supports)
the same conclusions
(or finds the same problems/conflicts)

“things” = methods, info, people,
processes, documents, ...

Triangulation

Elicitation Methods

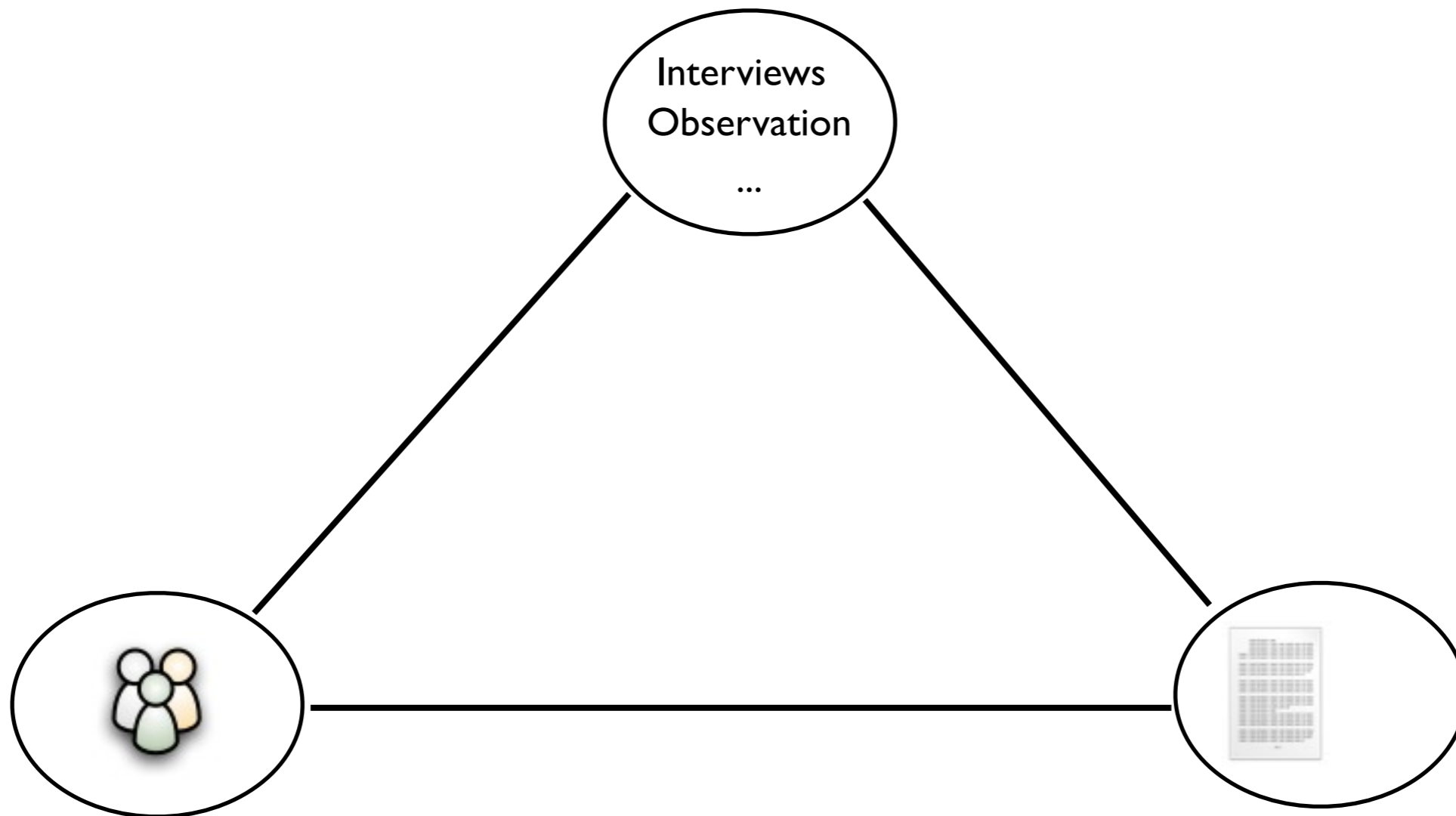


People / Stakeholders

Artifacts / Docs

Triangulation

Elicitation Methods

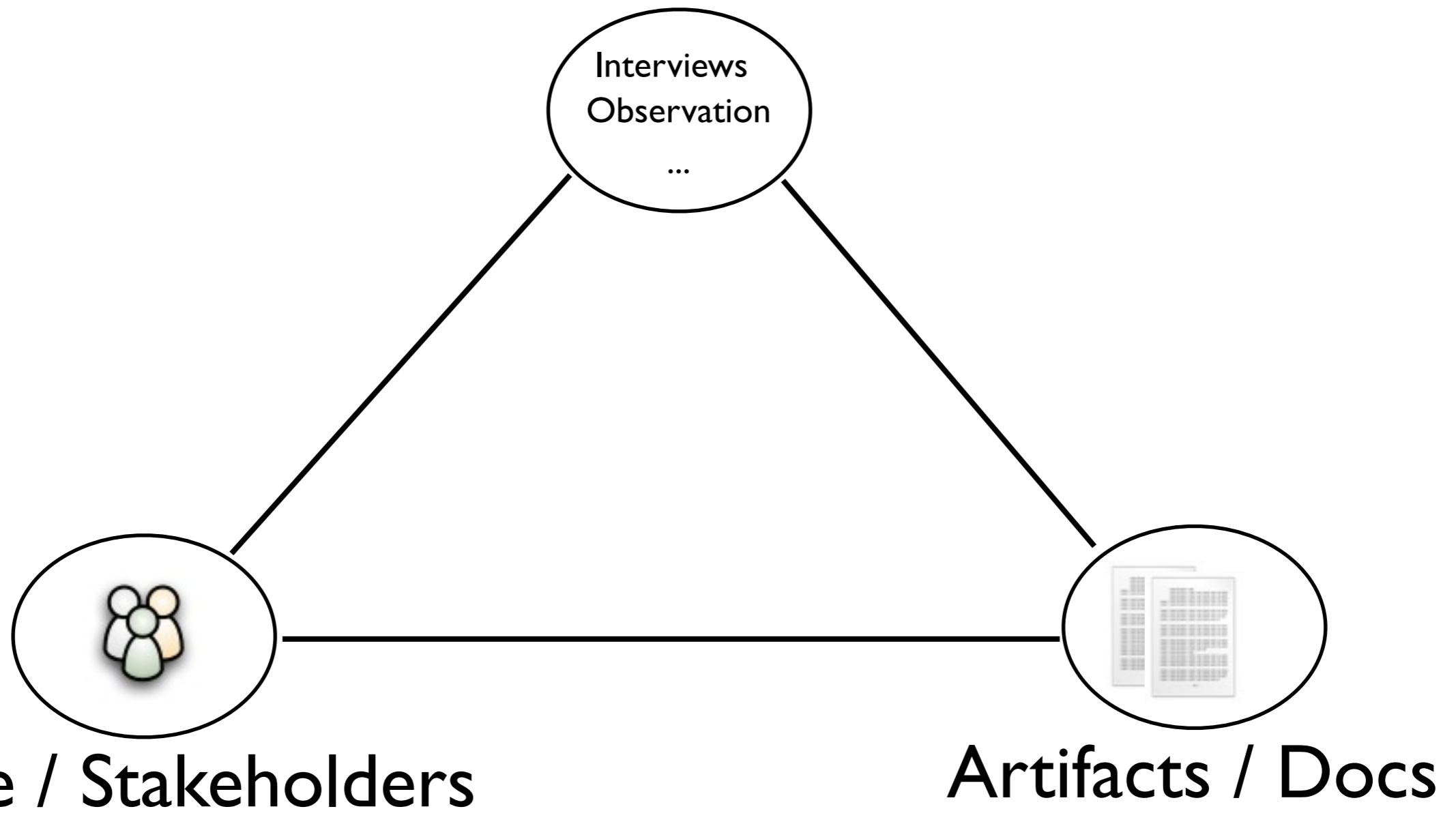


People / Stakeholders

Artifacts / Docs

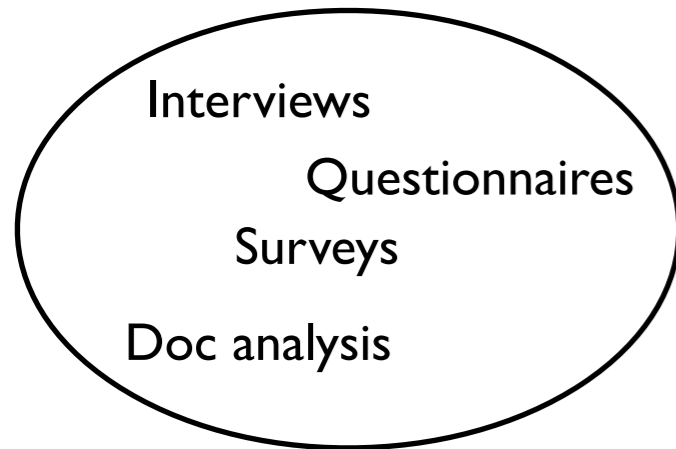
Triangulation

Elicitation Methods



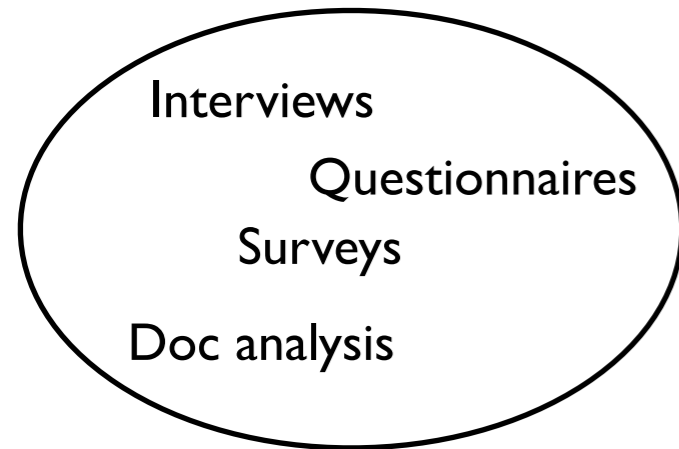
Elicitation methods

Elicitation methods

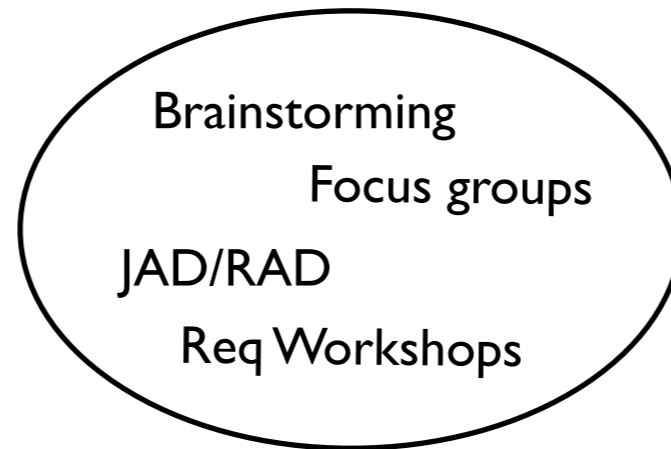


“Traditional”

Elicitation methods

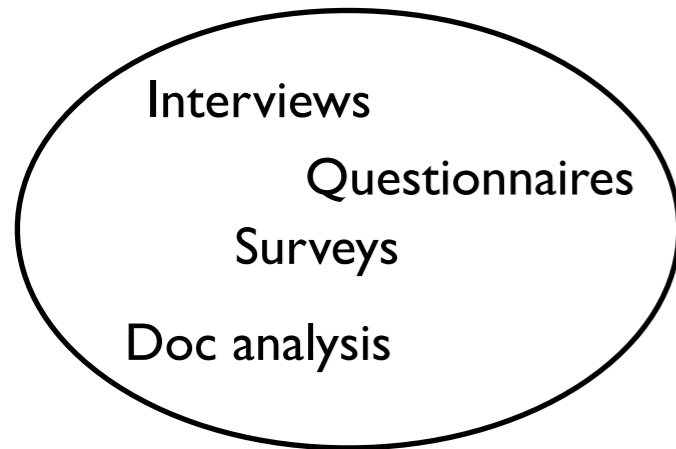


“Traditional”

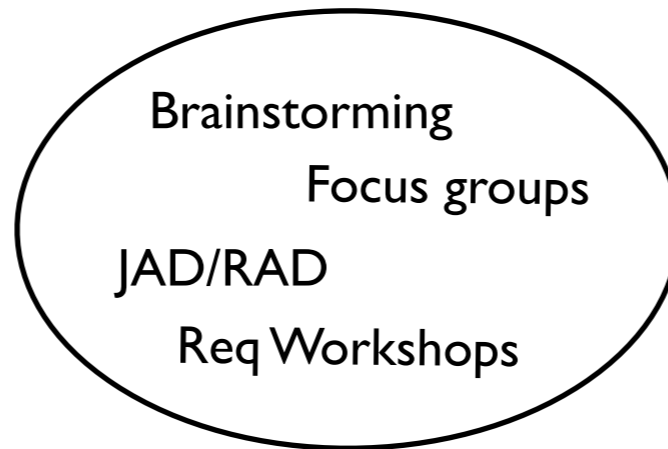


Group-based

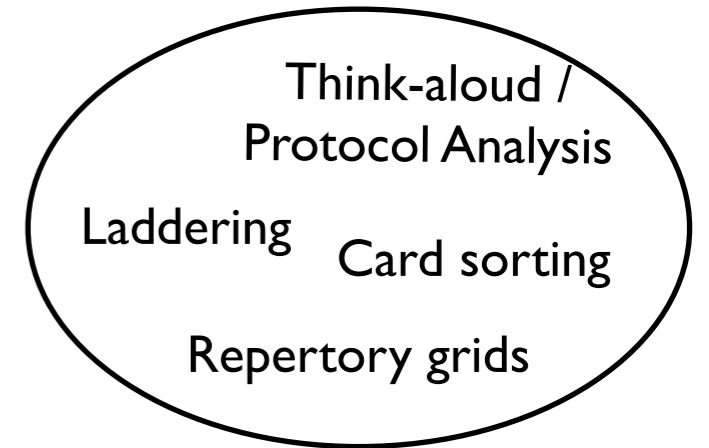
Elicitation methods



“Traditional”

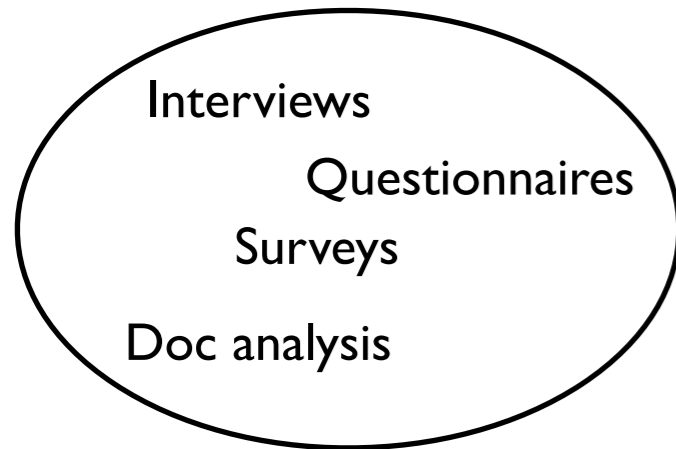


Group-based

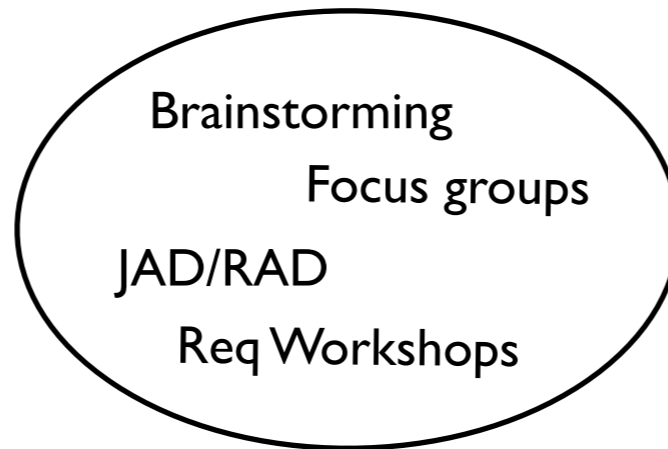


“Cognitive”

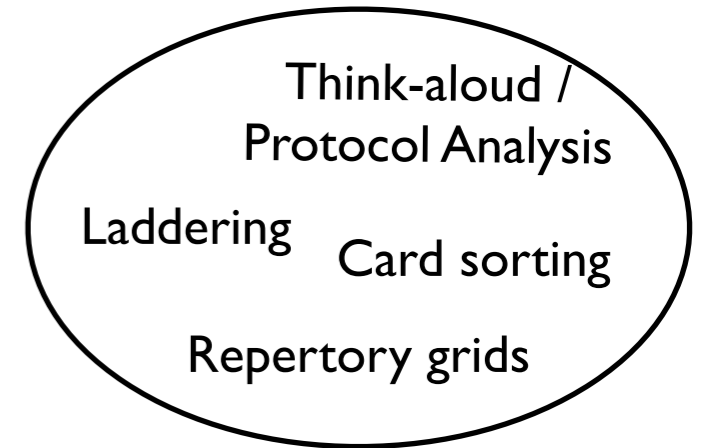
Elicitation methods



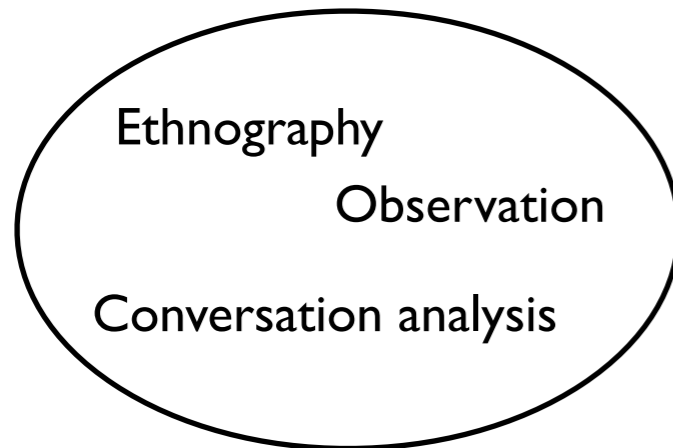
“Traditional”



Group-based

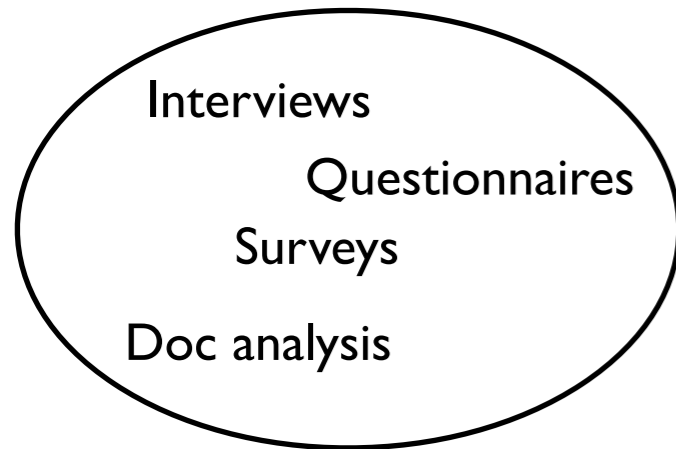


“Cognitive”

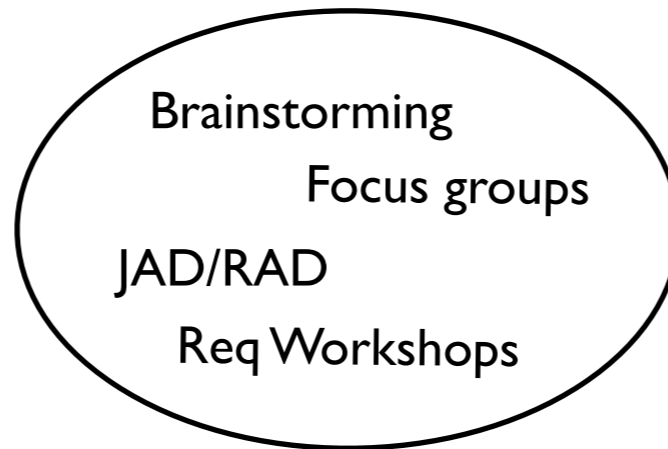


Contextual

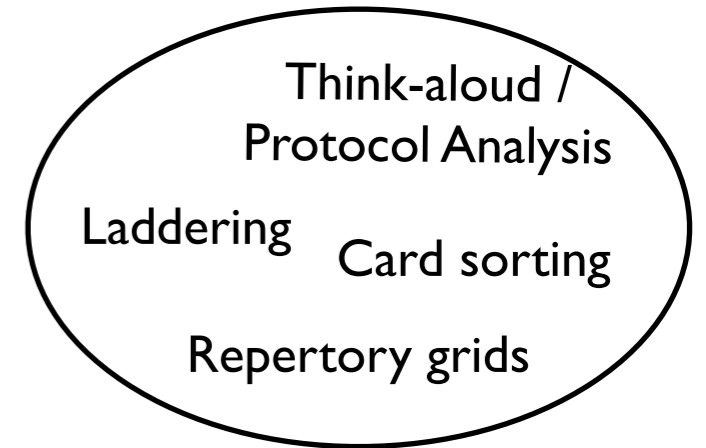
Elicitation methods



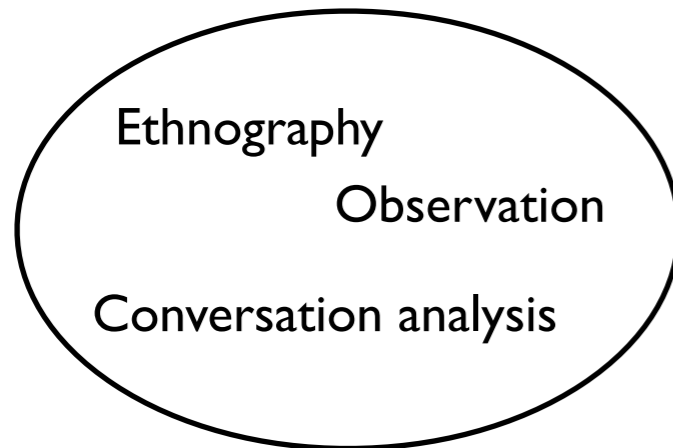
“Traditional”



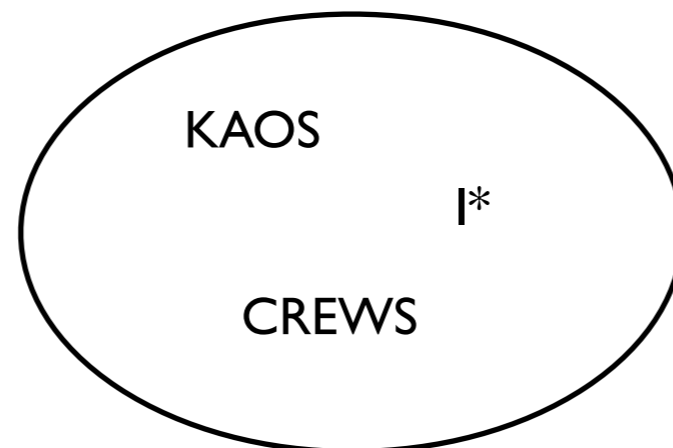
Group-based



“Cognitive”

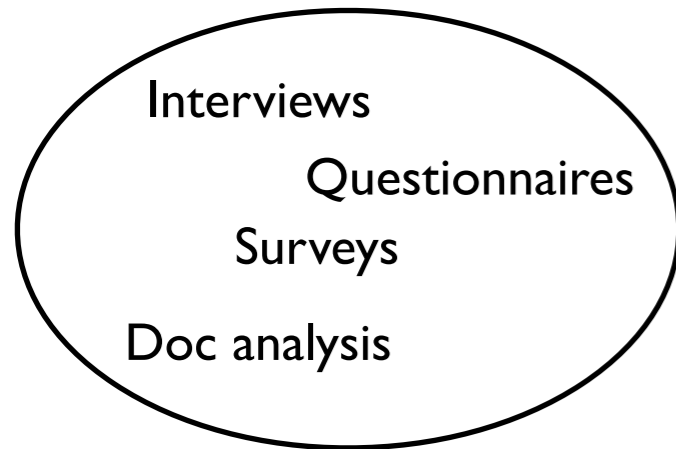


Contextual

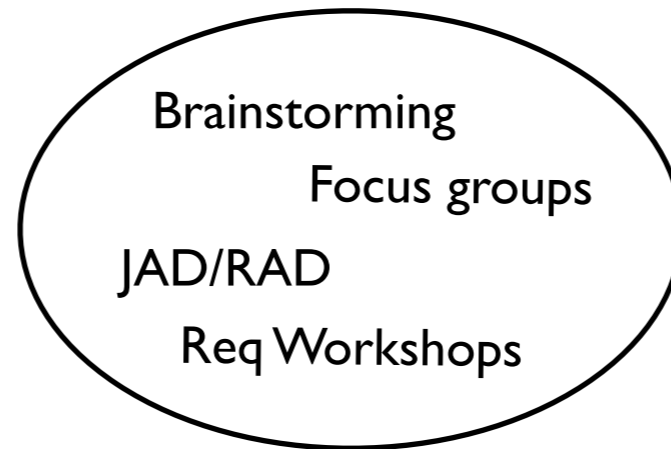


Model-driven

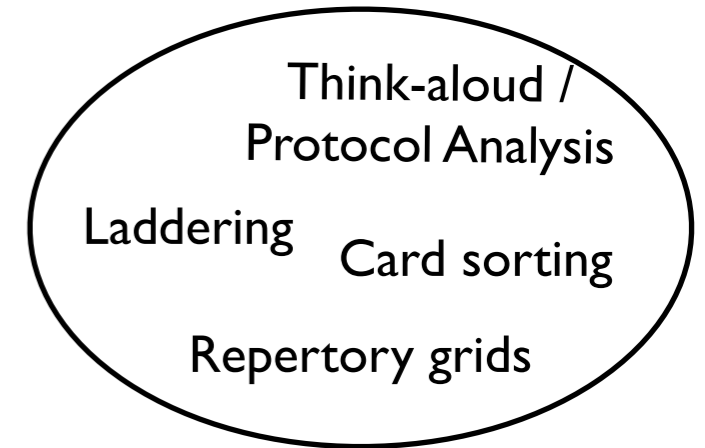
Elicitation methods



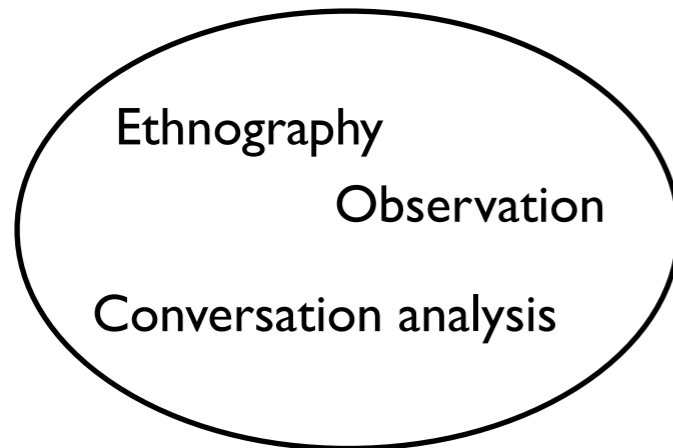
“Traditional”



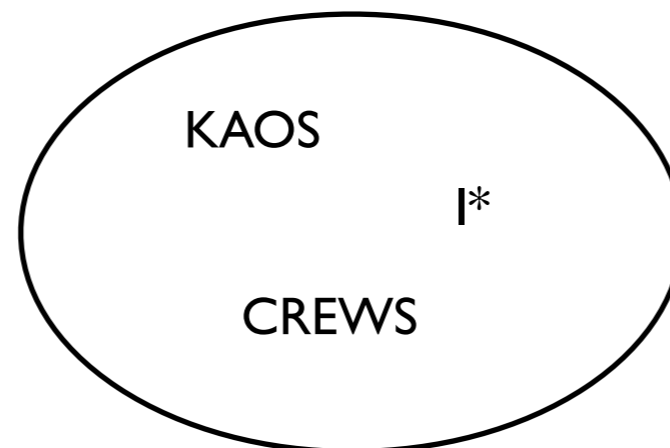
Group-based



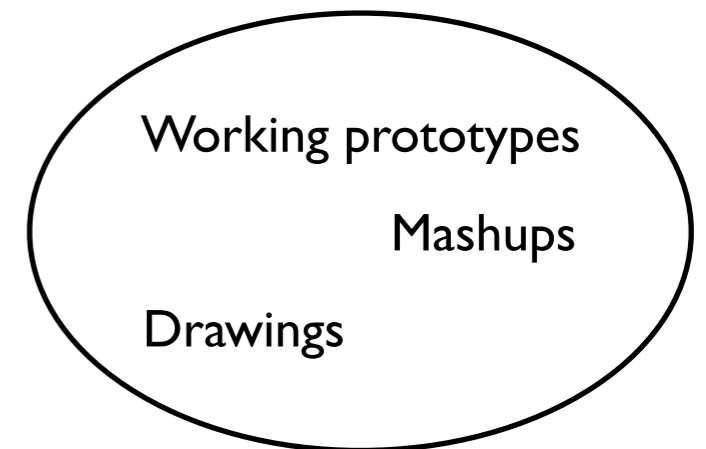
“Cognitive”



Contextual



Model-driven




Prototyping

Elicitation techniques - early

Technique	Pro	Con
Interviews	Know the present & future ideas, Uncover conflicts/politics	Goals & critical issues, Subjective
Group interviews/sessions	Stimulate/complete each other, Many/Diverse stakeholders	Censorship & domination, Groupthink
Observation	Actual current behavior, processes	Time consuming, misses exceptional/usability problems

Elicitation techniques - early

Technique	Pro	Con
Interviews	Know the present & future ideas, Uncover conflicts/politics	Goals & critical issues, Subjective
	Stimulate/complete each other, Many/diverse stakeholders	Censorship & domination, Groupthink
	Actual current behavior, processes	Time consuming, misses exceptional/usability problems

Elicitation techniques - early



Pro

Con

How the present & future ideas, Uncover conflicts/politics

Goals & critical issues, Subjective

Group interviews/sessions

Stimulate/complete each other, Many/Diverse stakeholders

Censorship & domination, Groupthink

Observation

Actual current behavior, processes

Time consuming, misses exceptional/usability problems

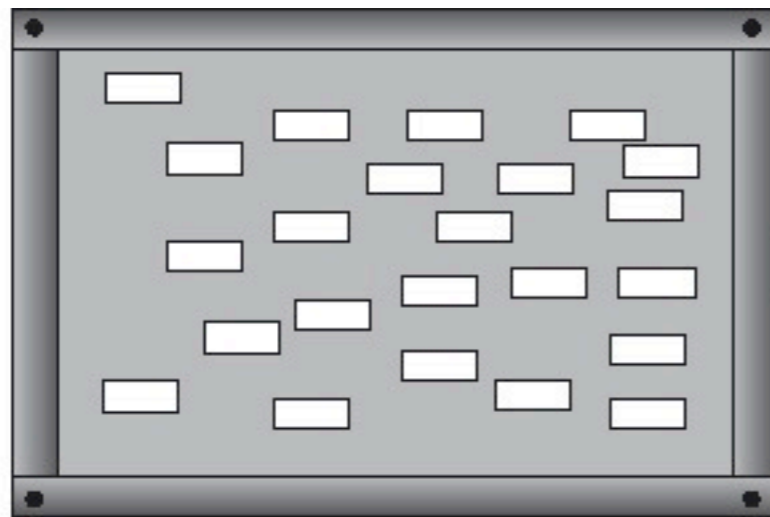
Elicitation techniques - mid

Technique	Pro	Con
Task demo	Clarify how work done	Presence & Qs influence, Critical issues seldom captured
Questionnaires	Info from many (statistics, views, opinions)	Hard to construct, Interpretation
Brainstorming	Many ideas (none rejected)	Many ideas (prioritization needed), Involvement

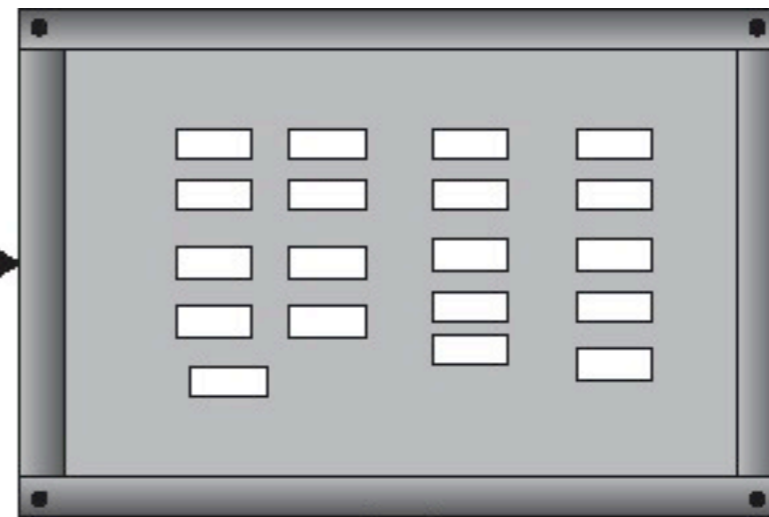
Elicitation techniques - late

Technique	Pro	Con
Use cases / Scenarios	Concentration on specifics => accuracy	Solution-oriented, Premature design
Modeling, Data-flow Diagrams, ...	Communication, Organize info, Uncover missing/ inconsistencies	Require tools, Time consuming, "Cults"
Prototyping	Visualization, Stimulate ideas, Usability centered	Solution-oriented, Premature design, "Already done?"

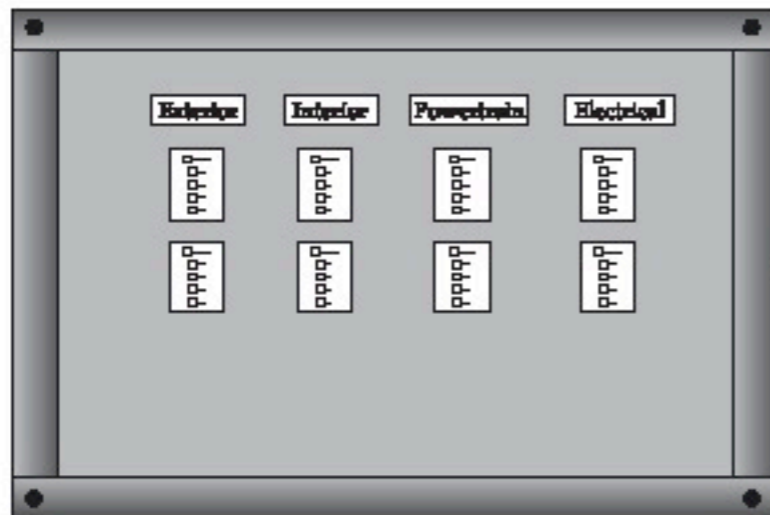
Brainstorming



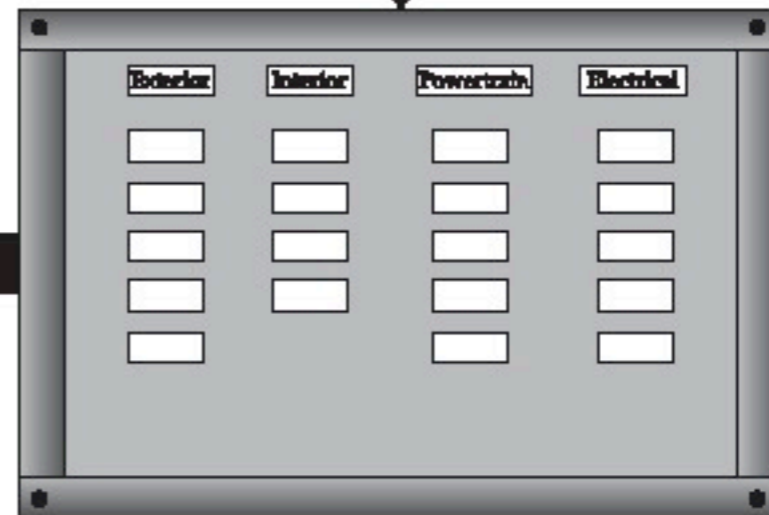
Unstructured Ideas



Grouped Related Ideas

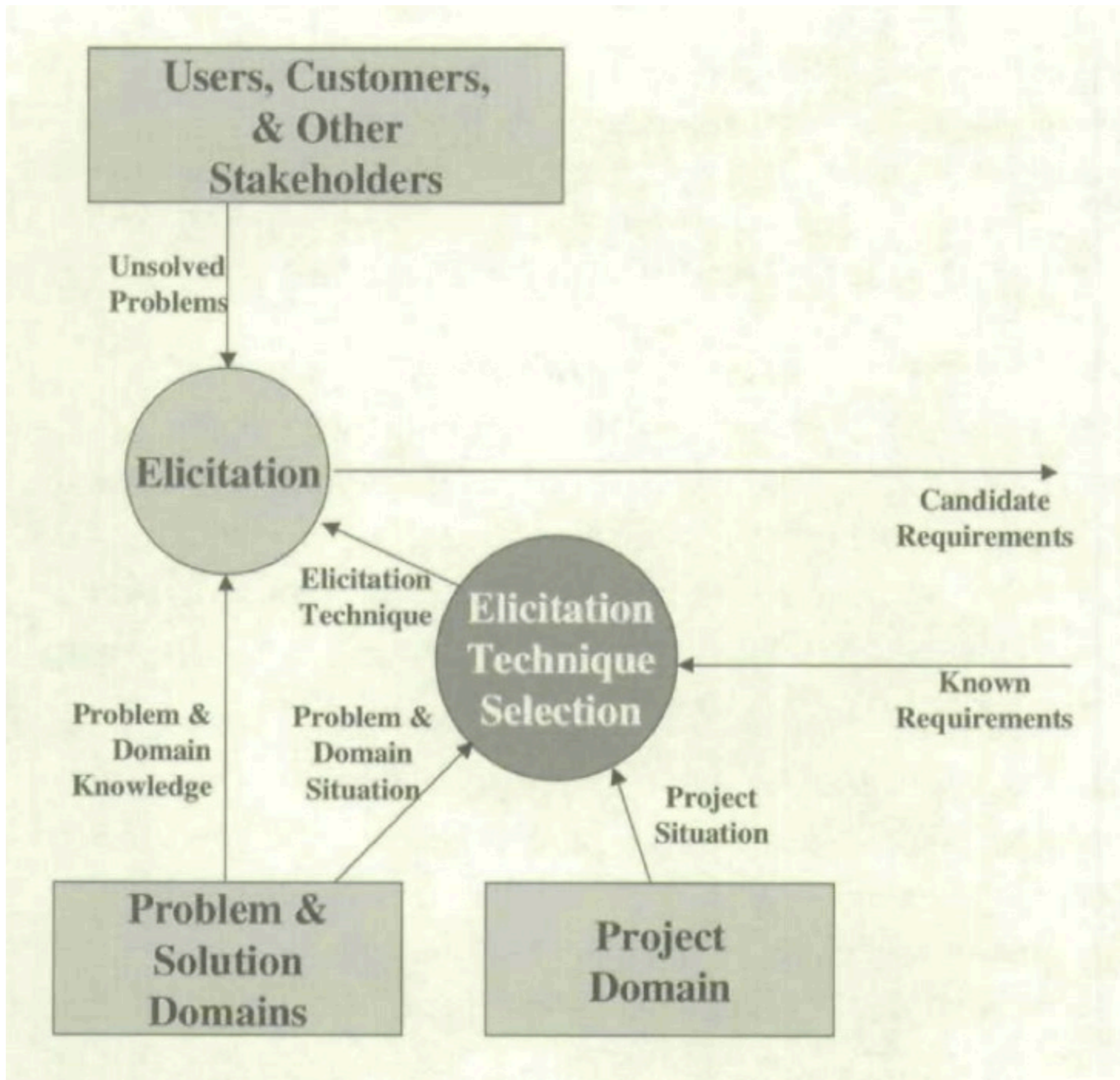


Expanded and Prioritized Ideas



Categorized Ideas





Research on how to elicit?

#	Aggregation result	(1)	(2)	Comments
1	Structured interviews gather more information than unstructured interviews	[3,11,63,67]	---	---
2	Unstructured interviews gather more information than sorting and ranking techniques	[10,16,20,80]	[5]	---
3	Unstructured interviews appear to gather more information than thinking aloud techniques	[13,16,20]	[22]	<ul style="list-style-type: none"> • The evidence given in [16] is confusing, but suggests that interviews are better than thinking aloud techniques. • The quality of the study [22] can be qualified as being on the low side
4	Elicitation techniques do not appear to provide specific types of information, that is, there is not enough evidence to support differential information access depending on what elicitation technique is used	[10,11,13,22,78]	[16]	<ul style="list-style-type: none"> • The quality of the study [22] can be qualified as being on the low side.
5	Analyst experience does not appear to be a relevant factor during information acquisition, at least using interviews as an elicitation technique.	[3,63,74]	[34]	---
6	The use of visual aids or prototypes focuses the discussion on the displayed artifact and does not generally help to discover new requirements.	[41,68]	---	<ul style="list-style-type: none"> • Not a lot of evidence is available as yet, although other studies (not covered by this review), like [30], support this finding.

Strategies for elicitation

Strategy	Description
Scenario Building	Asking a user to imagine or construct a scenario in his domain, and respond as he would in that situation
Conditionalizing	Use “if-then” to limit or clarify applicability of an assertion
Elaborating with examples	Asking a user to illustrate a point by providing examples
Hedging	Asking a user to design contingency plans or fallback positions

Strategies for elicitation

Strategy	Description
Scenario Building	Asking a user to imagine or construct a scenario in his domain and respond as he would in that situation “Describe the most unusual customer you ever had. How did you respond in that situation?”
Conditionalizing	Use “if-then” to limit or clarify applicability of an assertion
Elaborating with examples	Asking a user to illustrate a point by providing examples
Hedging	Asking a user to design contingency plans or fallback positions

Strategies for elicitation

Strategy	Description
Scenario Building	Asking a user to imagine or construct a scenario in his domain, and respond as he would in that situation
Conditionalizing	“If the project is finished as planned, then what does that mean for the customer?”
Elaborating with examples	Asking a user to illustrate a point by providing examples
Hedging	Asking a user to design contingency plans or fallback positions

Strategies for elicitation

Strategy	Description
Scenario Building	Asking a user to imagine or construct a scenario in his domain, and respond as he would in that situation
Conditionalizing	Use “if-then” to limit or clarify applicability of an assertion
Elaborating with examples	“Can you provide some examples of what you mean?”
Hedging	Asking a user to design contingency plans or fallback positions

Strategies for elicitation

Strategy	Description
Scenario Building	Asking a user to imagine or construct a scenario in his domain, and respond as he would in that situation
Conditionalizing	Use “if-then” to limit or clarify applicability of an assertion
Elaborating with examples	Asking a user to illustrate a point by providing examples
Hedging	“What would you do if this action would not give the desired result?”

Strategies for elicitation

Strategy	Description
Scenario Building	Asking a user to imagine or construct a scenario in his domain, and respond as he would in that situation
Conditionalizing	Use “if-then” to limit or clarify applicability of an assertion
Elaborating with examples	Asking a user to illustrate a point by providing examples
Hedging	Asking a user to design contingency plans or fallback positions

Strategies for elicitation

Strategy	Description
Generating Counterargument	Asking a stakeholder to argue against the conclusion she first reached
Generating Arguments	Asking for more or different arguments favoring a position
Feedback	Asking for or giving feedback, either verbally or in writing / on notes
Summarization	Asking for or giving a summary

Strategies for elicitation

Strategy	Description
Generating Counterargument	“Why might the system not work as well as you say it will?”
Generating Arguments	Asking for more or different arguments favoring a position
Feedback	Asking for or giving feedback, either verbally or in writing / on notes
Summarization	Asking for or giving a summary

Strategies for elicitation

Strategy	Description
Generating Counterargument	Asking a stakeholder to argue against the conclusion she first reached
Generating Arguments	“Can you think of an analogy that would help clarify what you are saying?”
Feedback	Asking for or giving feedback, either verbally or in writing / on notes
Summarization	Asking for or giving a summary

Strategies for elicitation

Strategy	Description
Generating Counterargument	Asking a stakeholder to argue against the conclusion she first reached
Generating Arguments	Asking for more or different arguments favoring a position
Feedback	<p>“Let me recap what I have noted down from our conversation and you can see if you agree?”</p>
Summarization	Asking for or giving a summary

Strategies for elicitation

Strategy	Description
Generating Counterargument	Asking a stakeholder to argue against the conclusion she first reached
Generating Arguments	Asking for more or different arguments favoring a position
Feedback	Asking for or giving feedback, either verbally or in writing / on notes
Summarization	

Strategies for elicitation

Strategy	Description
Generating Counterargument	Asking a stakeholder to argue against the conclusion she first reached
Generating Arguments	Asking for more or different arguments favoring a position
Feedback	Asking for or giving feedback, either verbally or in writing / on notes
Summarization	Asking for or giving a summary

Task Characteristics Prompting

What would your customers want the system to do?
Substantive Prompt

Why would your customers not want to use the system?
Procedural Prompt—Causal Counterargument

What can be done to overcome these negatives?
Procedural Prompt—Causal Counterargument

What would your employees want the system to do?
Substantive Prompt

Summarize everything you want the system to do.
Procedural Prompt—Summarization, Feedback

What must the customer do to use the system?
Substantive Prompt

What must the employees do to use the system?
Substantive Prompt

Can you think of a situation in which the customer would have a problem using the system?
Procedural Prompt—Scenario Building

What can be done to overcome these problems?
Procedural Prompt—Casual Counterargument

Summarize the steps for using the system.
Procedural Prompt—Summarization, Feedback

What people or departments must be involved to support the customer's use of the system?
Substantive Prompt

What people or departments must be involved to support the employees' use of the system?
Substantive Prompt

Describe in detail the tasks that these people or departments must do.
Substantive Prompt

What feedback must the system provide to assist in performing these tasks?
Substantive Prompt

Can you think of a situation in which the customer would have to make a decision or choice when using the system?
Procedural Prompt—Scenario Building

What kinds of things can people do now that they might not be able to do when using the system?
Procedural Prompt—Casual Counterargument

What information must a customer supply to the system to be able to use it?
Substantive Prompt

What information must the system supply to the customer?
Substantive Prompt

What information must the employees supply to the system to be able to use it?
Substantive Prompt

What information must the system supply to the employees?
Substantive Prompt

Semantic Prompting

Goals

- What are the system goals?
- How is each goal attained?
- Why is each goal important?
- What indicates that each goal is achieved?

Agents

- Can you name a person or department involved with the system?
- What role does each play?
- What are his or her goals?
- What agent has opposing goals?

Actions

- Can you name the actions involved in the system?
- How does a person perform each action?
- What prevents a person from being able to perform each action?
- What goal(s) does each action satisfy?

Events

- What events affect the system?
- What are the consequences of each event occurring?
- What causes each event to occur?
- What goal does each event fulfill?

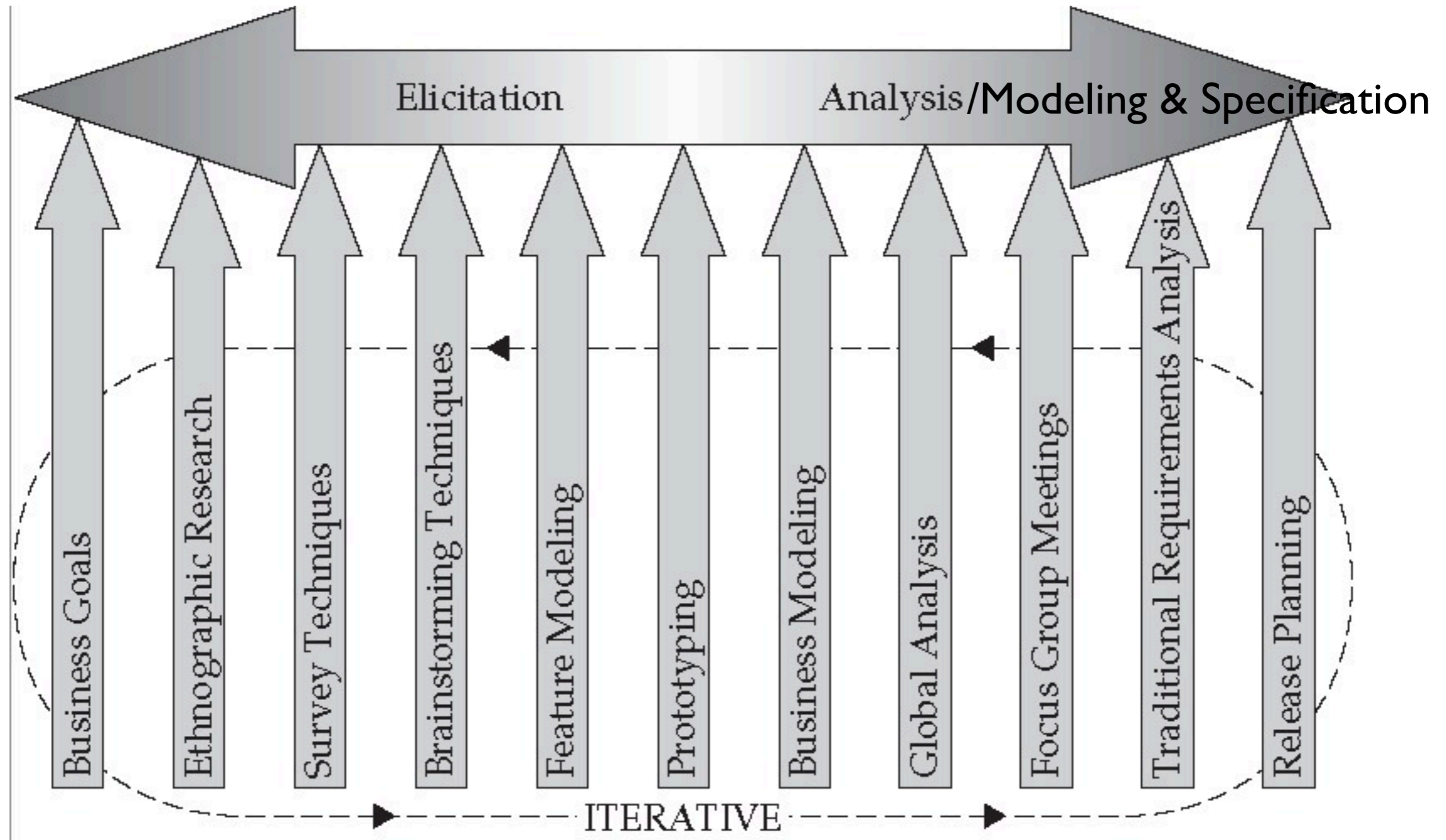
States or Conditions

- What states or conditions affect the system?
- What causes or enables each state?
- What are the consequences of each state being present?
- What goal does each state support?

A question to ponder:

Can you think of an elicitation situation where you would choose to start elicitation with hand-drawn UI sketches or is that never good early?

A continuum



What is Req Specification?

What is Req Specification?

*“The deliberate **documentation** of requirements to a **degree** that makes the associated **risks tolerable**”*

What is Req Specification?

*“The deliberate **documentation** of requirements to a **degree** that makes the associated **risks tolerable**”*

i.e. writing requirements down in a form so that we avoid later problems

What is Req Modeling?

What is Req Modeling?

*“The construction of
abstract descriptions of
reqs/goals/systems/behavior”*

What is Req Modeling?

*“The construction of
abstract descriptions of
reqs/goals/systems/behavior”*

*Used in several RE activities:
elicitation, analysis, specification*

What are risks without doc?

- Reqs still ambiguous & open-ended after elicitation =>
- Developers make decisions/assumptions later =>
- User <-> Dev difference: User not satisfied
- Dev <-> Dev difference: Inconsistent system
- Overall: Costs high!
- BUT:
 - Goal is ideal PRODUCT not ideal Req Doc!
 - Thus: Just enough Req Spec to reduce Risks!