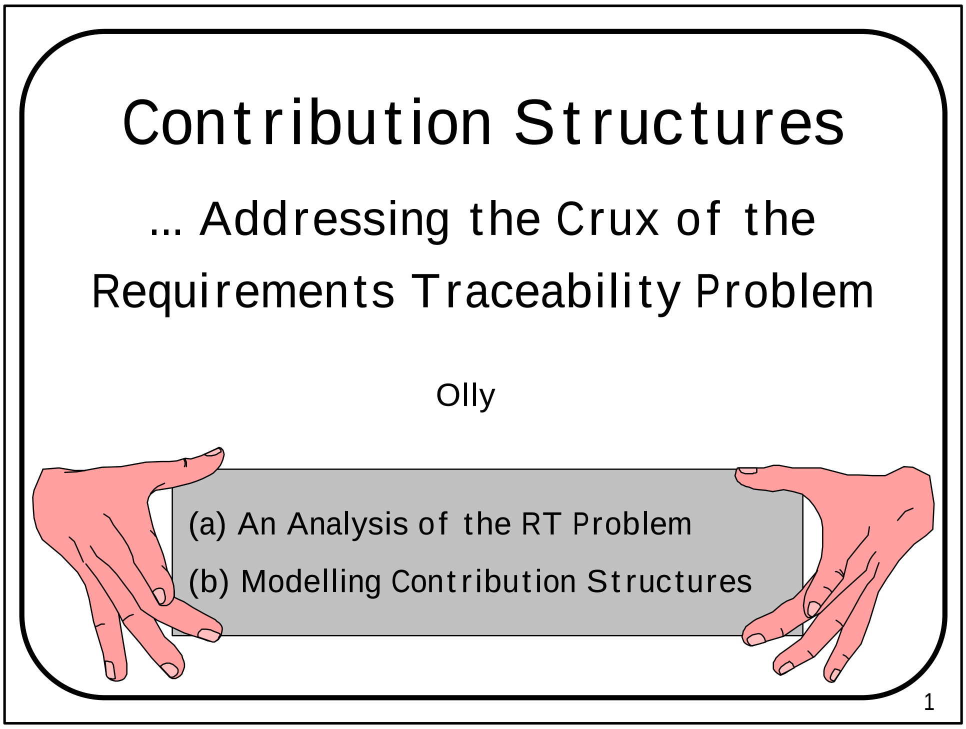


Contribution Structures

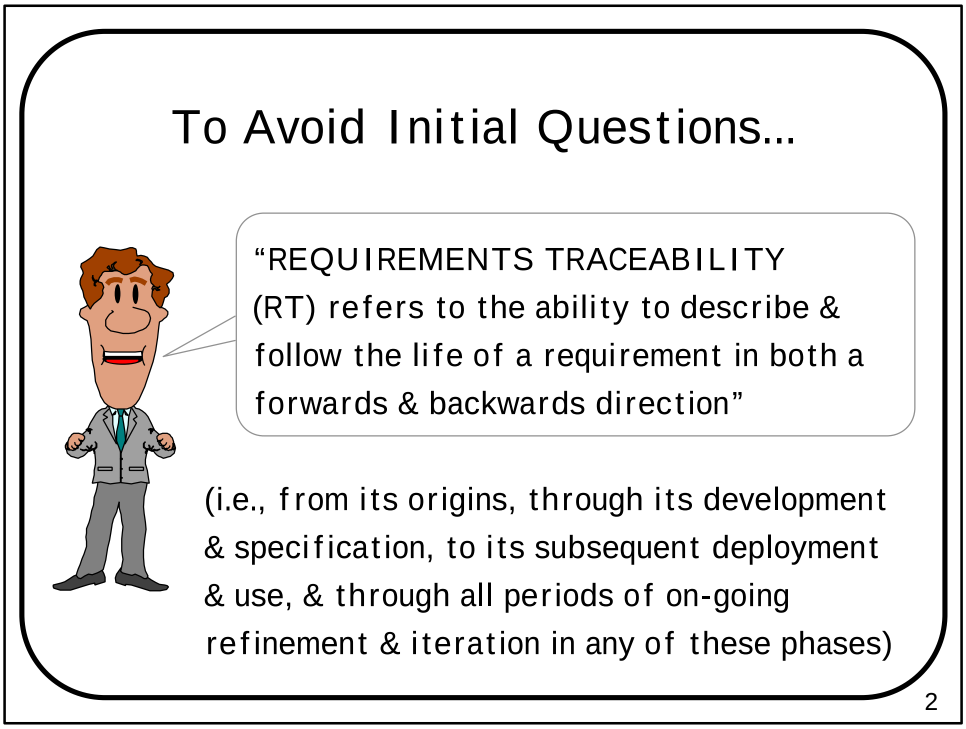
... Addressing the Crux of the Requirements Traceability Problem

Olly

(a) An Analysis of the RT Problem

(b) Modelling Contribution Structures

1

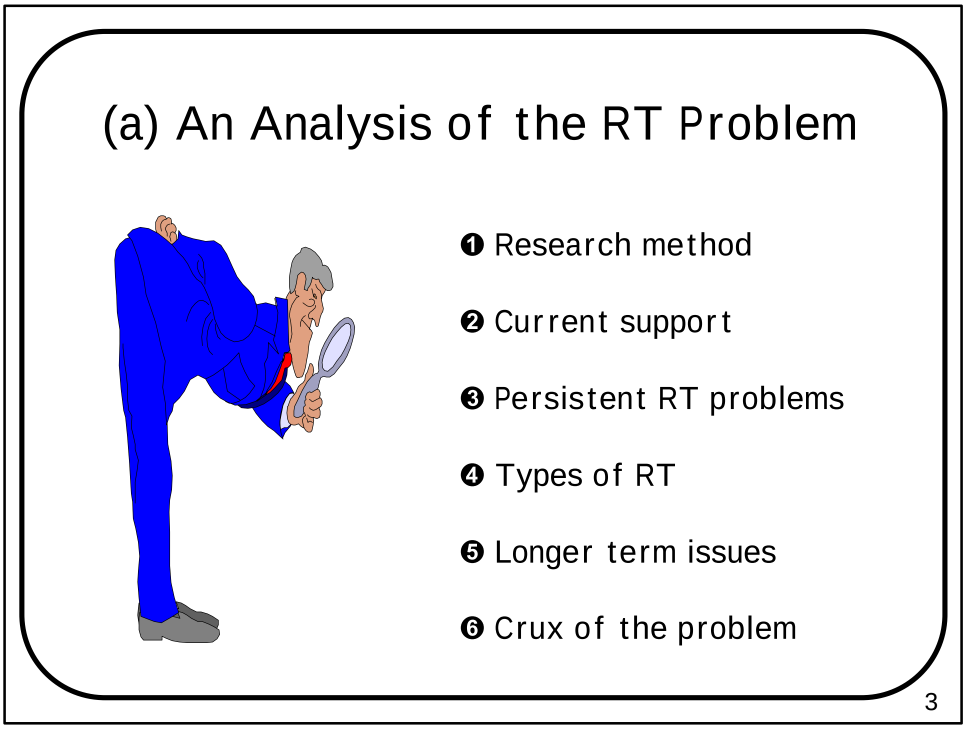


To Avoid Initial Questions...

"REQUIREMENTS TRACEABILITY (RT) refers to the ability to describe & follow the life of a requirement in both a forwards & backwards direction"

(i.e., from its origins, through its development & specification, to its subsequent deployment & use, & through all periods of on-going refinement & iteration in any of these phases)

2



(a) An Analysis of the RT Problem

❶ Research method

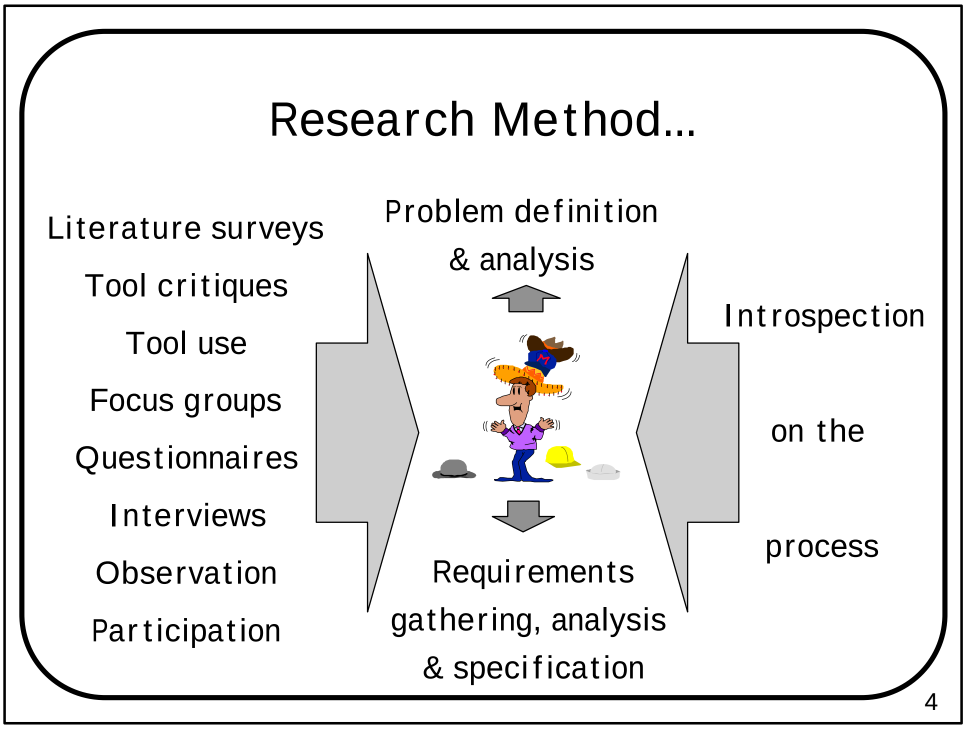
❷ Current support

❸ Persistent RT problems

❹ Types of RT

❺ Longer term issues

❻ Crux of the problem



Research Method...

Literature surveys
Tool critiques
Tool use
Focus groups
Questionnaires
Interviews
Observation
Participation

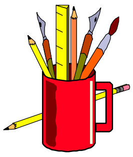
Problem definition & analysis

Requirements gathering, analysis & specification

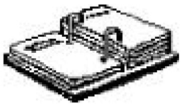
Introspection on the process

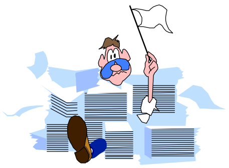
Current Support - Mechanics...

- Explicit techniques:
  - ☞ Cross reference schemes / Matrices
  - ☞ Templates / Documents
  - ☞ ATMS / Constraint networks



- Implicit approcahes:
  - ☞ Languages
  - ☞ Models
  - ☞ Methods

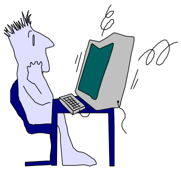


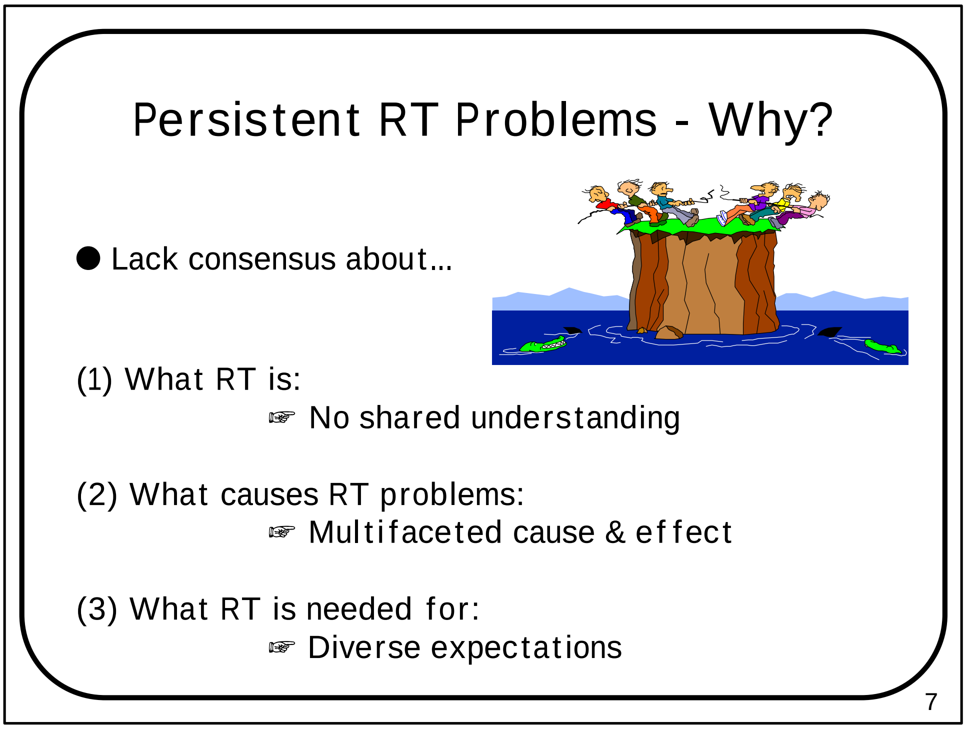


Current Support - Tools...

- General purpose tools
- Special purpose tools

- Workbenches:
  - ☞ Dedicated to requirements
  - ☞ Conventional upper & lower CASE
- Environments (& beyond):
  - ☞ Language-based
  - ☞ Structure-based
  - ☞ Method-based
  - ☞ Toolkit-based





Persistent RT Problems - Why?

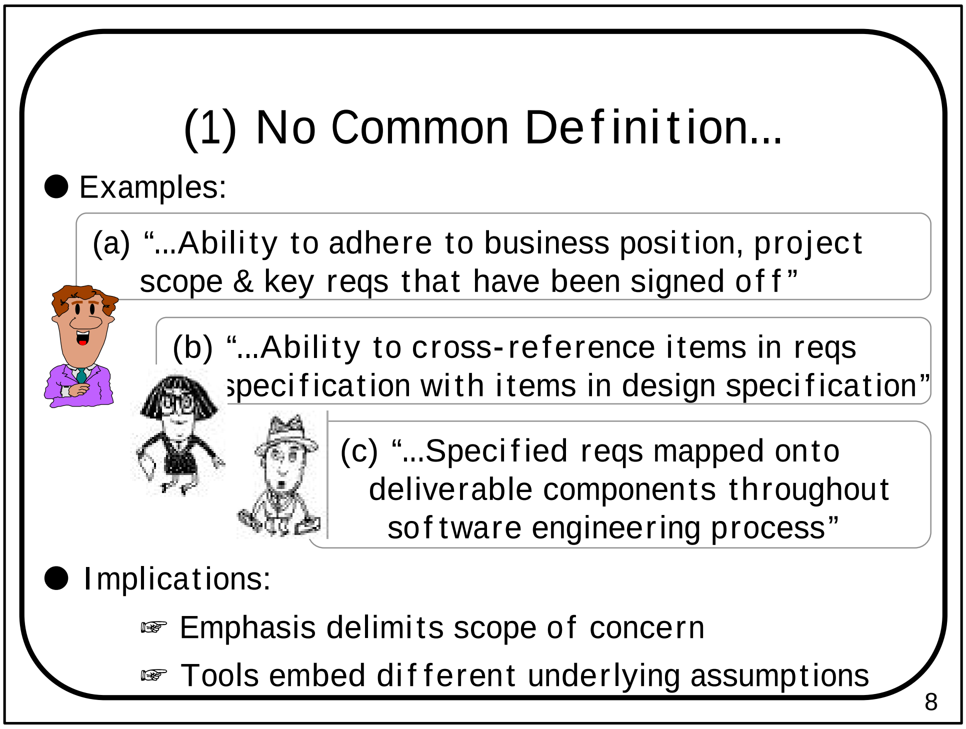
- Lack consensus about...

(1) What RT is:
☞ No shared understanding

(2) What causes RT problems:
☞ Multifaceted cause & effect

(3) What RT is needed for:
☞ Diverse expectations

7



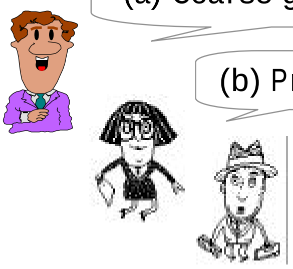
(1) No Common Definition...

- Examples:
  - (a) "...Ability to adhere to business position, project scope & key reqs that have been signed off"
  - (b) "...Ability to cross-reference items in reqs specification with items in design specification"
  - (c) "...Specified reqs mapped onto deliverable components throughout software engineering process"
- Implications:
  ☞ Emphasis delimits scope of concern
  ☞ Tools embed different underlying assumptions

8

(2) Multiple Problem Causes...

- Examples:



(a) Coarse granularity of traceable entities

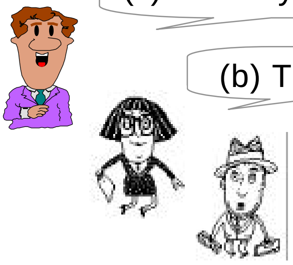
(b) Project longevity

(c) Lack of commitment by all parties

- Implications:
  - ☞ Problem statement ambiguity
  - ☞ Tools address different underlying problems

(3) Numerous Expectations...

- Examples:

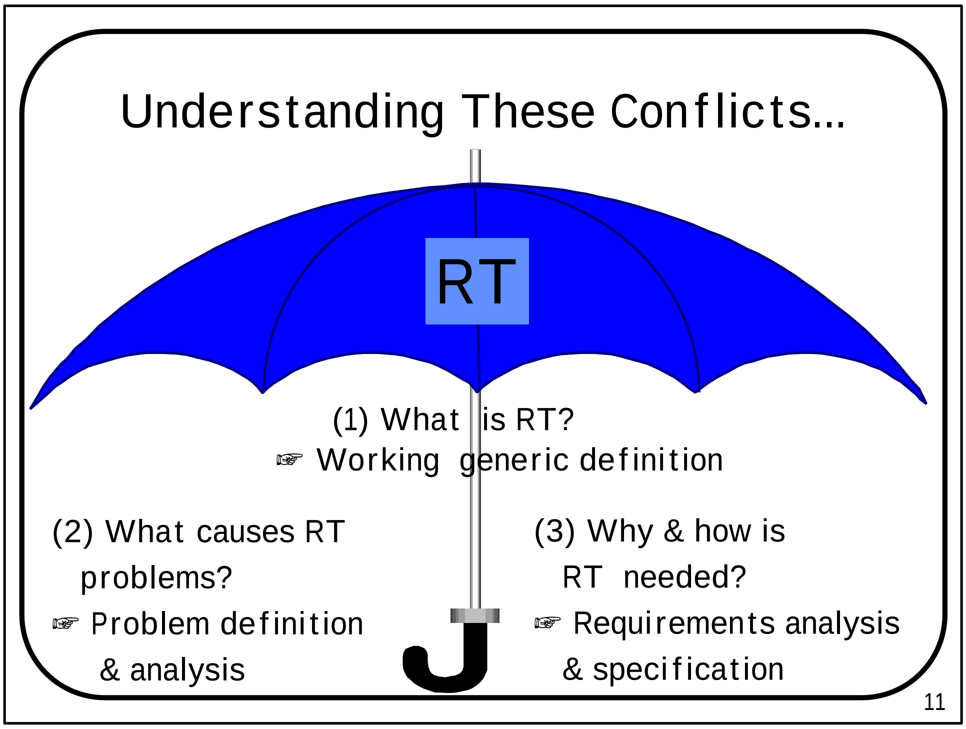


(a) To analyse consistency & completeness

(b) To assess change impact

(c) To see requirements from multiple viewpoints

- Implications:
  - ☞ Unclear (user) requirements for RT
  - ☞ Limitations on what RT can achieve



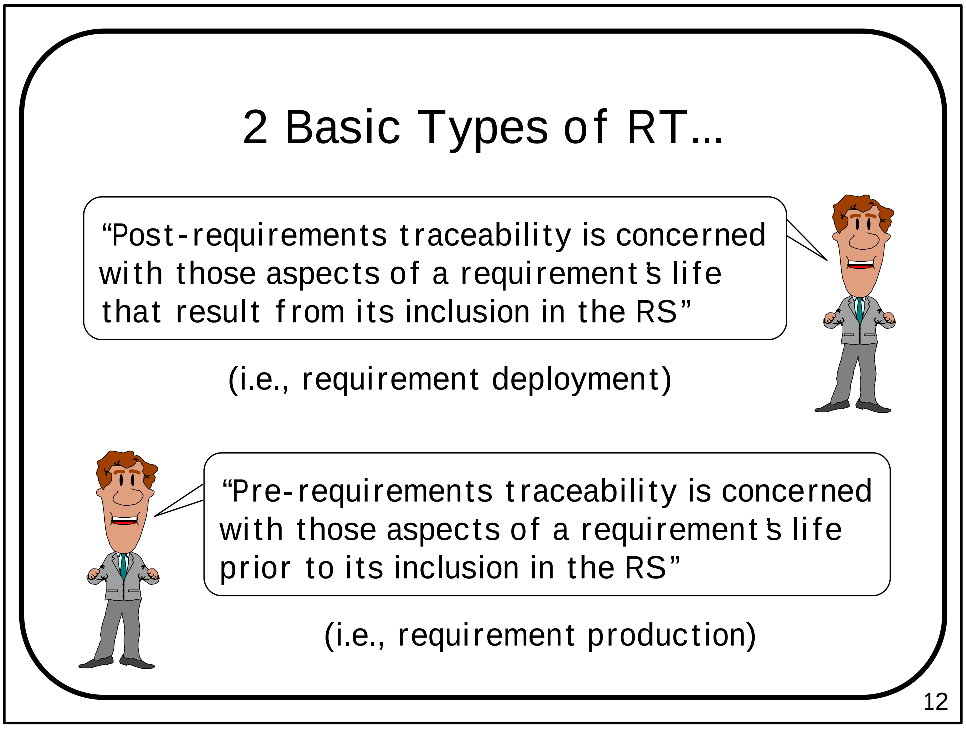
Understanding These Conflicts…

RT

(1) What is RT?
☞ Working generic definition

(2) What causes RT problems?
☞ Problem definition & analysis

(3) Why & how is RT needed?
☞ Requirements analysis & specification



2 Basic Types of RT…

"Post-requirements traceability is concerned with those aspects of a requirement's life that result from its inclusion in the RS"

(i.e., requirement deployment)

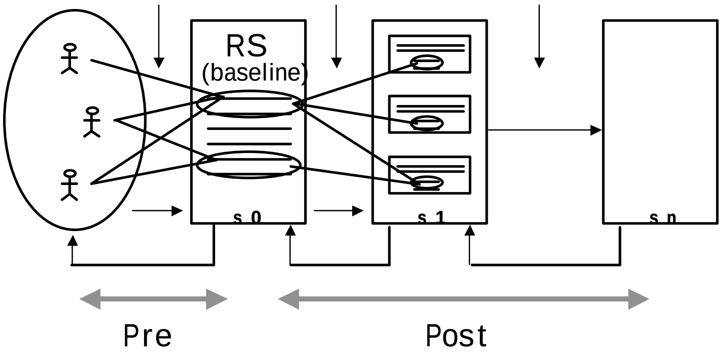
"Pre-requirements traceability is concerned with those aspects of a requirement's life prior to its inclusion in the RS"

(i.e., requirement production)

Pre & Post-Reqs Traceability…

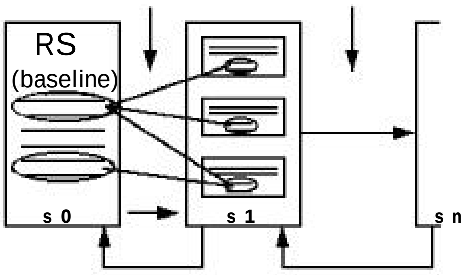
Reqs artifacts produced
(related to production)

Reqs artifacts produced
(related to deployment)



Post-Reqs Traceability…

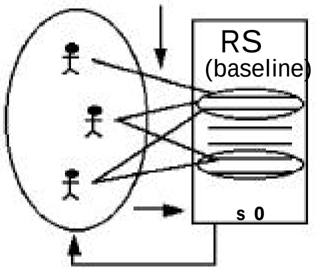
Intermediate artifacts



- Well understood & supported
- Remaining problems tackled in formal settings
- Limited impact on reducing problems

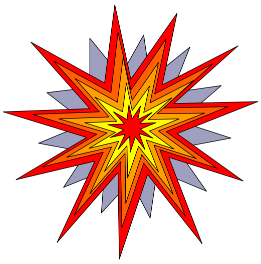
Pre-Reqs Traceability...

Intermediate artifacts



- ● Poorly understood & supported
- ● Only contributor to problems in formal settings
- ● Instrumental in reducing long term problems

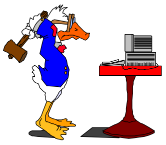
Work Tackling Pre-Reqs Issues...



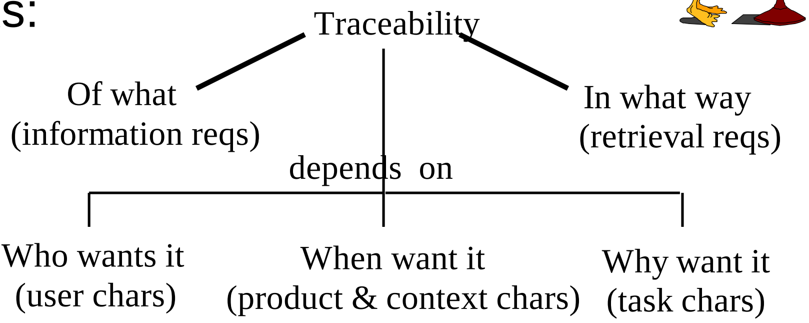
- ● Awareness of requirements:
  - ☞ Frameworks & activity models / Common threads of involvement
- ● Obtaining & recording:
  - ☞ RE tools / Exploratory workbenches
- ● Organising & maintaining:
  - ☞ Requirements as modular viable systems / Roles
- ● Accessing & representing:
  - ☞ Programmability / Context-sensitive dynamic traces

But - RT Reqs Are Situated...

Comprehensive & up to date project information + Sophisticated retrieval & presentation ≠ No RT problems



As:



Traceability

Of what (information reqs)

In what way (retrieval reqs)

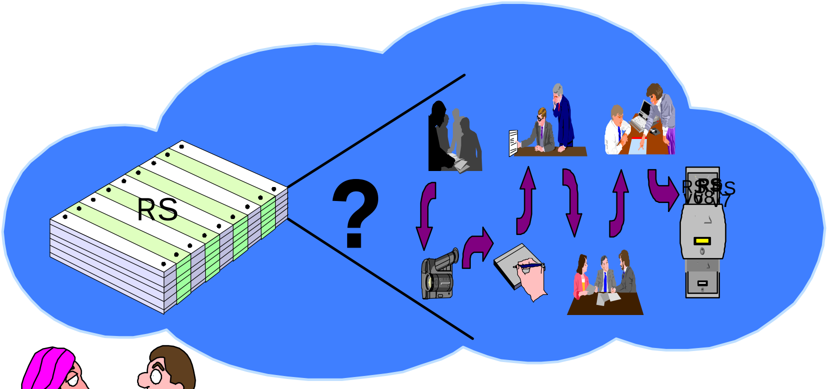
depends on

Who wants it (user chars)

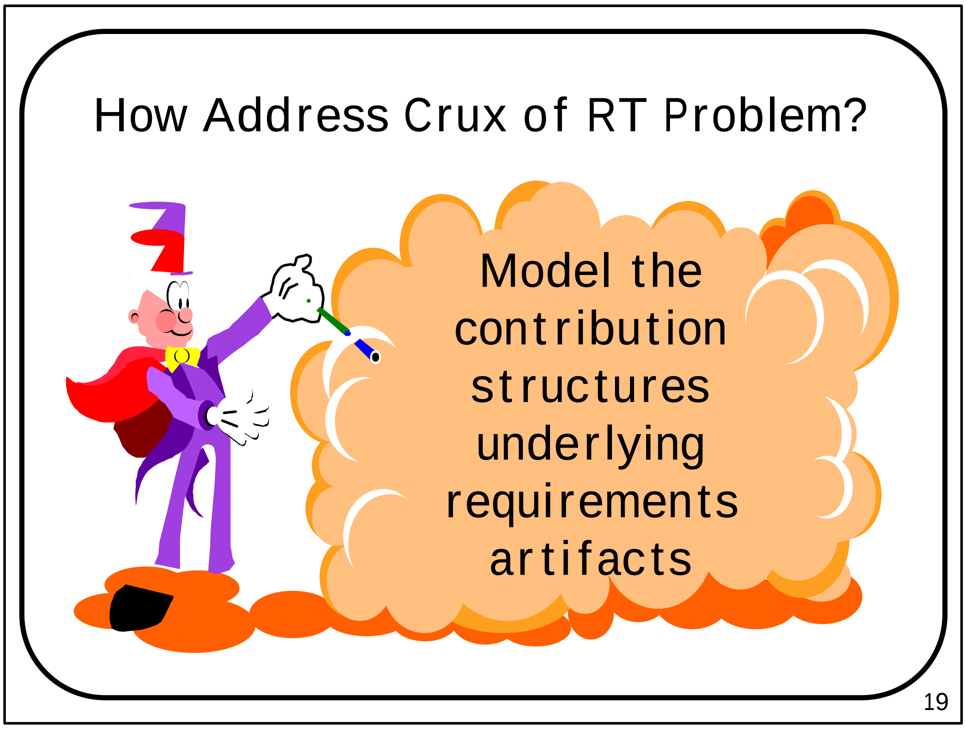
When want it (product & context chars)

Why want it (task chars)

A Fundamental Working Practice Is...

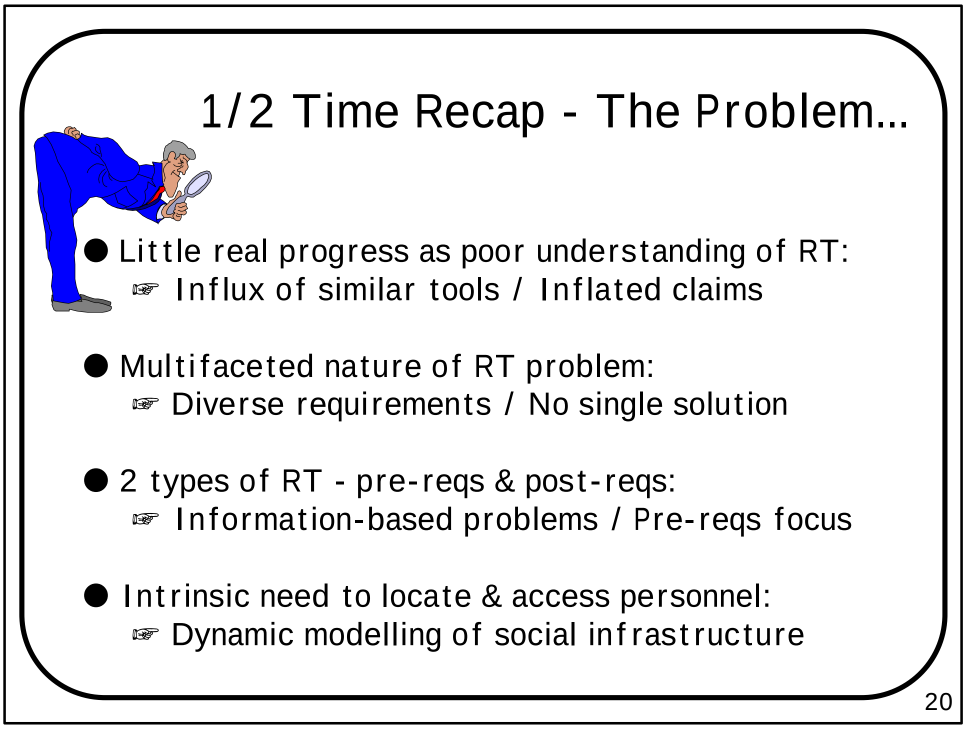


- ● Location & access of personnel
  - ☞ To back up / To augment



How Address Crux of RT Problem?

Model the contribution structures underlying requirements artifacts



1/2 Time Recap - The Problem...

- Little real progress as poor understanding of RT:
  - ☞ Influx of similar tools / Inflated claims
- Multifaceted nature of RT problem:
  - ☞ Diverse requirements / No single solution
- 2 types of RT - pre-reqs & post-reqs:
  - ☞ Information-based problems / Pre-reqs focus
- Intrinsic need to locate & access personnel:
  - ☞ Dynamic modelling of social infrastructure

(b) Modelling Contribution Structures

❶ Outline of the approach

❷ Relating agents & artifacts

❸ Relating artifacts

❹ Roles & commitments

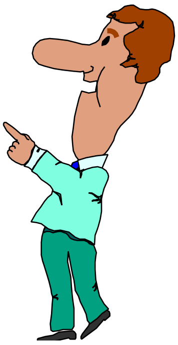
❺ Implementation

❻ Scenario

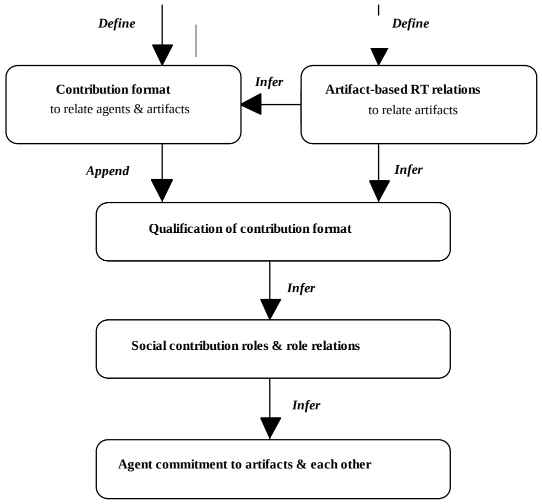
❼ Discussion

Some Preliminaries...

- Social infrastructure
- Scope of concern, problems to address & assumptions
- Requirements:
  - ☞ Differentiate how agents contribute
  - ☞ Account for artifact-based relations
  - ☞ Basis for modelling & reasoning
- Contribution structure
- Insightful areas



Outline of the Approach...



Define

Define

Contribution format
to relate agents & artifacts

Infer

Artifact-based RT relations
to relate artifacts

Append

Infer

Qualification of contribution format

Infer

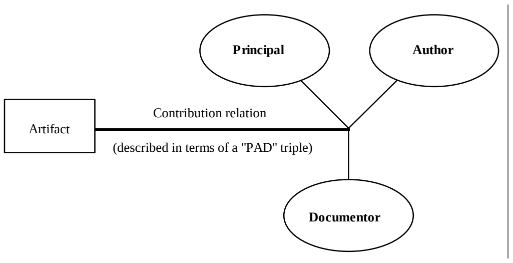
Social contribution roles & role relations

Infer

Agent commitment to artifacts & each other

Relating Agents & Artifacts...

- Goffman's "participant roles" ➜ Contribution format



Principal

Author

Artifact

Contribution relation

(described in terms of a "PAD" triple)

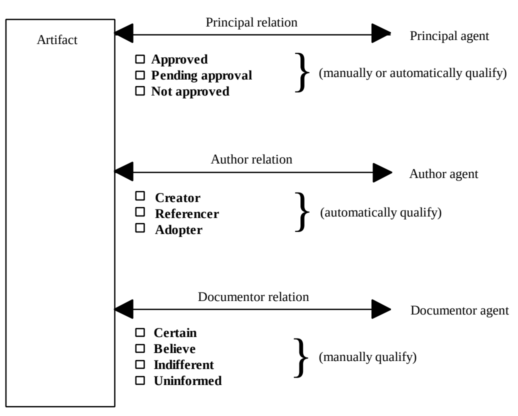
Documentor

P: agent whose position/belief is established

A: agent who formulated/organised content & structure

D: agent who recorded or transcribed

Qualifying Contribution Relations…



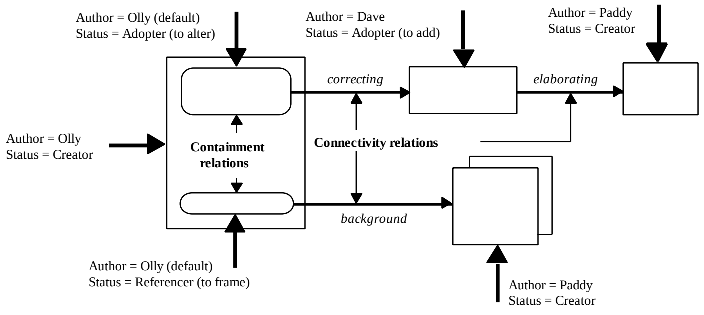
- ● Signatures
- ● Sources
- ● Moods

Relating Artifacts…

① Temporal ② Developmental ③ Auxiliary

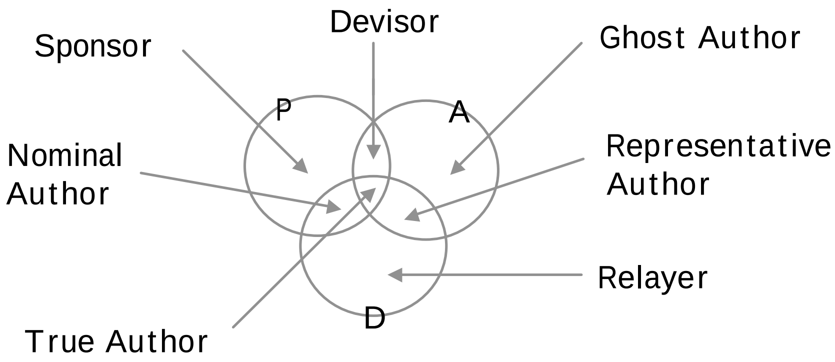
Containment
(layering & nesting)

Connectivity
(cohesion & coherence)

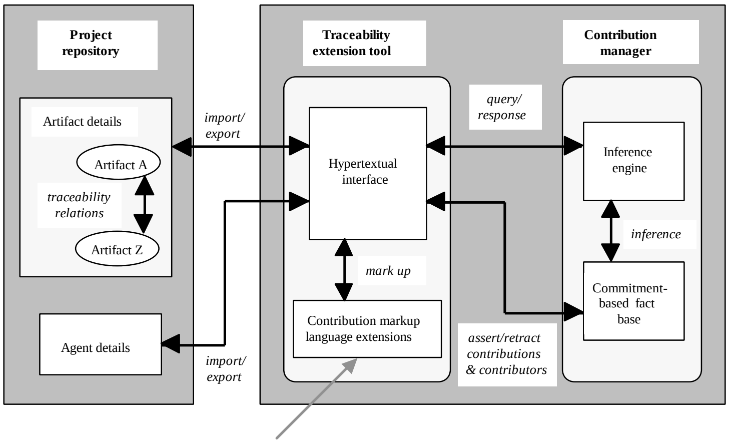


Roles & Commitments...

- Individual & collective commitment to artifacts
- Social commitment to each other - role relations



Implementation...



<CP=[Agent, Qualification]> ... </<CP>
<REL=[Source, Target, Function, Purpose]>
etc.....

Part of the Underlying Model...

Basic types :
[AGENT]
[ARTIFACT]

Data type definitions :
CAPACITY ::= Principal | Author | Documentor
RELATION ::= Contains | References | Adopts
QUALIFICATION ::= PQUALIF | AQUALIF | DQUALIF
PQUALIF ::= Approved | Pendingapproval | Notapproved
AQUALIF ::= Creator | Referencer | Adopter
DQUALIF ::= Certain | Believe | Indifferent | Uninformed
PURPOSE ::= CPURPOSE | RPURPOSE | APURPOSE
CPURPOSE ::= Component
RPURPOSE ::= Frame | Match | Substantiate | Causal
APURPOSE ::= Copy | Add | Remove | Alter

Derived sets:
REGISTERED_AGENT: P AGENT
REGISTERED_ARTIFACT: P ARTIFACT
GROUP_AGENT: P (AGENT X P AGENT)
CONTRIBUTION_RELATION: P (AGENT X ARTIFACT X CAPACITY X QUALIFICATION)
ARTIFACT_RELATION: P (ARTIFACT X ARTIFACT X RELATION X PURPOSE)

State:
REGISTERED_AGENT U REGISTERED_ARTIFACT U GROUP_AGENT U CONTRIBUTION_RELATION U ARTIFACT_RELATION

Some Things Made Possible...

all_agents_and_their_contributions : P (AGENT X P ARTIFACT)
== {ag : AGENT; art_ list: setARTIFACT | ∀art : ARTIFACT • art ∈ art_ list ⇒ is_ contributor_ to (art , ag)}

agent_collaborates_on_artifacts_with(ag) : AGENT --> P AGENT
== {ag_ list: setAGENT | ∀ a : AGENT • a ∈ ag_ list ⇒ (ag_ contributions (ag) ∩ ag_ contributions (a)) ≠ ∅}

agent_has_related_agents(ag) : AGENT --> P AGENT
== ag_ collabs_ on_ arts_ with (ag) ∪ ag_ group_ membs (ag) ∪ ag_ membs_ of(ag) ∪ ag_ membs_ with (ag)

mediating_artifact(art1, art2) : ARTIFACT X ARTIFACT--> BOOLEAN
=> (((art_related_ arts_ thro_ in_relations (art 1) ∩ art_related_ arts_ thro_ out_relations (art 2)) ≠ ∅) ∨
((art_related_ arts_ thro_ in_relations (art 2) ∩ art_related_ arts_ thro_ out_relations (art 1)) ≠ ∅))

mediating_agent(art1, art2) : ARTIFACT X ARTIFACT--> BOOLEAN
=> (∃ art 3 : ARTIFACT s. t. mediating_ artifact (art 1, art 2) ∧ (((art_ contributors (art 1) ∩
art_ contributors (art 3)) ≠ ∅) ∧ ((art_ contributors (art 2) ∩ art_ contributors (art 3)) ≠ ∅))

contributors_to_id_artifacts(ag1, ag2) : AGENT X AGENT --> BOOLEAN
=> agent_ contributions (ag 1) ≡ agent_ contributions (ag 2)

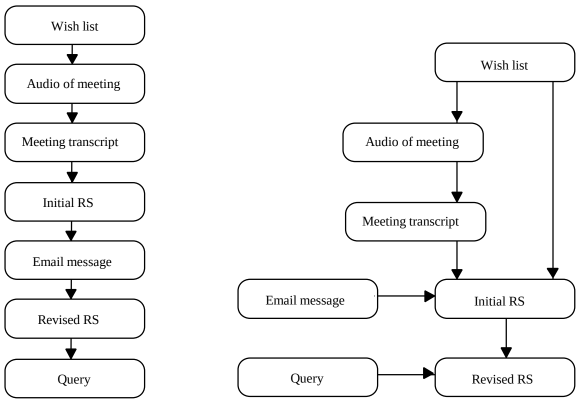
related_to_id_agents(ag1, ag2): AGENT X AGENT --> BOOLEAN
=> contributors_ to_ id_ artifacts (ag 1, ag 2) ∧ members_ of _ id_ groups (ag 1, ag 2)

Consider a Scenario...

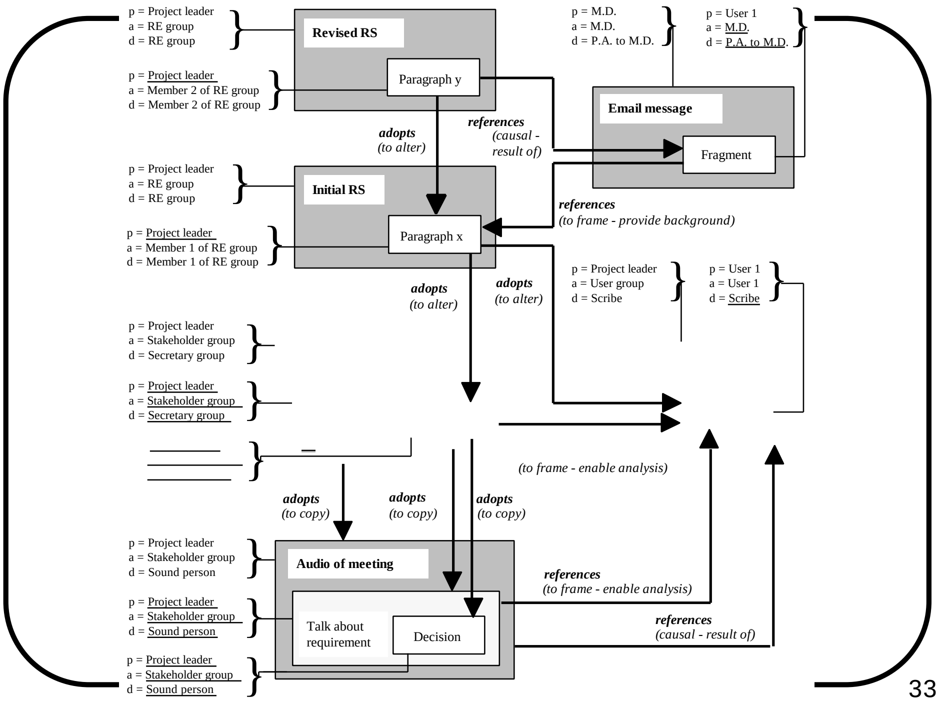
A software project began with a wish list, reporting needs from a **user group**, written up by a **scribe** and authorised by a **project leader**. The project leader then held a meeting, of which an audio tape record was made, to discuss the wish list with **stakeholders**. A direct transcript of the meeting was subsequently made by some **secretaries**. From the transcript and wish list, along with other input documents, an initial RS was written by a **group of requirements engineers**. Following circulation to and comments from interested parties, a revised version of the RS was written. In particular, an alteration had been made to the requirement covered by paragraph x, as a result of an email message from the M.D.'s **P.A.** to the project leader. In this message, the **M.D.** passed on a verbal change request she received from **user 1**. The changed version of paragraph x becomes paragraph y in the revised RS. Unfortunately, **member 2** of the requirements engineers introduced an error when carrying out this change, largely because he did not acknowledge the subtlety of the wording in the fragment of the email.

31

Artifact Chronology & Flow-Down...



32



p = Project leader
a = RE group
d = RE group

Revised RS

Paragraph y

p = Project leader
a = Member 2 of RE group
d = Member 2 of RE group

p = M.D.
a = M.D.
d = P.A. to M.D.

p = User 1
a = M.D.
d = P.A. to M.D.

Email message

Fragment

adopts *(to alter)*

references *(causal - result of)*

p = Project leader
a = RE group
d = RE group

Initial RS

Paragraph x

p = Project leader
a = Member 1 of RE group
d = Member 1 of RE group

references *(to frame - provide background)*

adopts *(to alter)*

adopts *(to alter)*

p = Project leader
a = User group
d = Scribe

p = User 1
a = User 1
d = Scribe

p = Project leader
a = Stakeholder group
d = Secretary group

p = Project leader
a = Stakeholder group
d = Secretary group

(to frame - enable analysis)

adopts *(to copy)*

adopts *(to copy)*

adopts *(to copy)*

p = Project leader
a = Stakeholder group
d = Sound person

Audio of meeting

Talk about requirement

Decision

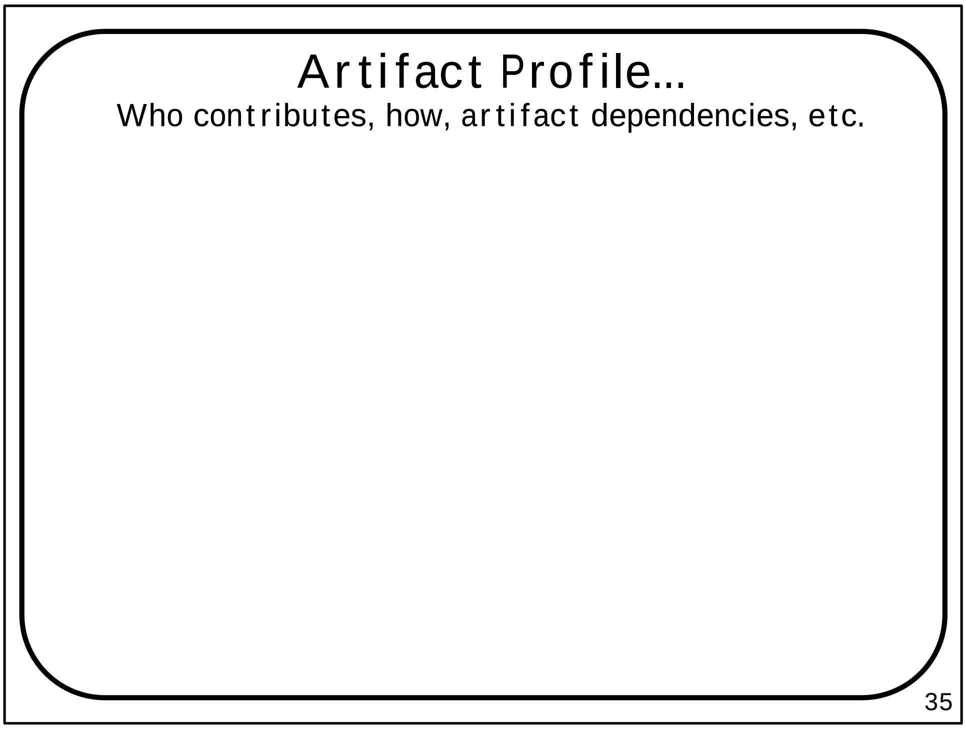
p = Project leader
a = Stakeholder group
d = Sound person

p = Project leader
a = Stakeholder group
d = Sound person

references *(to frame - enable analysis)*

references *(causal - result of)*

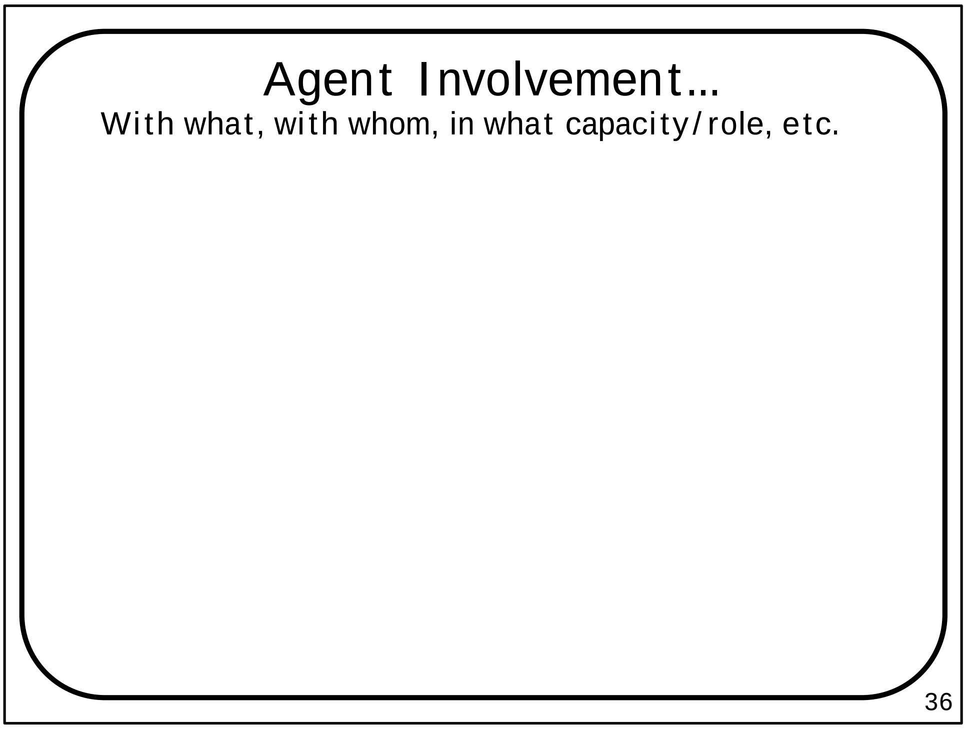
Original Source & Contributors...



Artifact Profile...

Who contributes, how, artifact dependencies, etc.

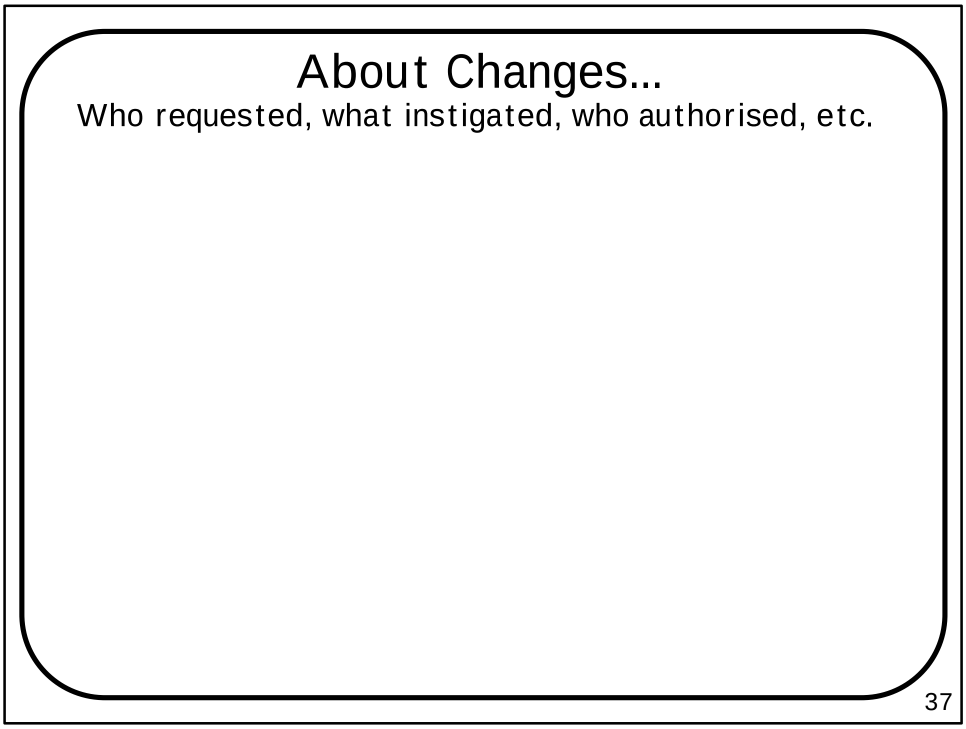
35



Agent Involvement...

With what, with whom, in what capacity/role, etc.

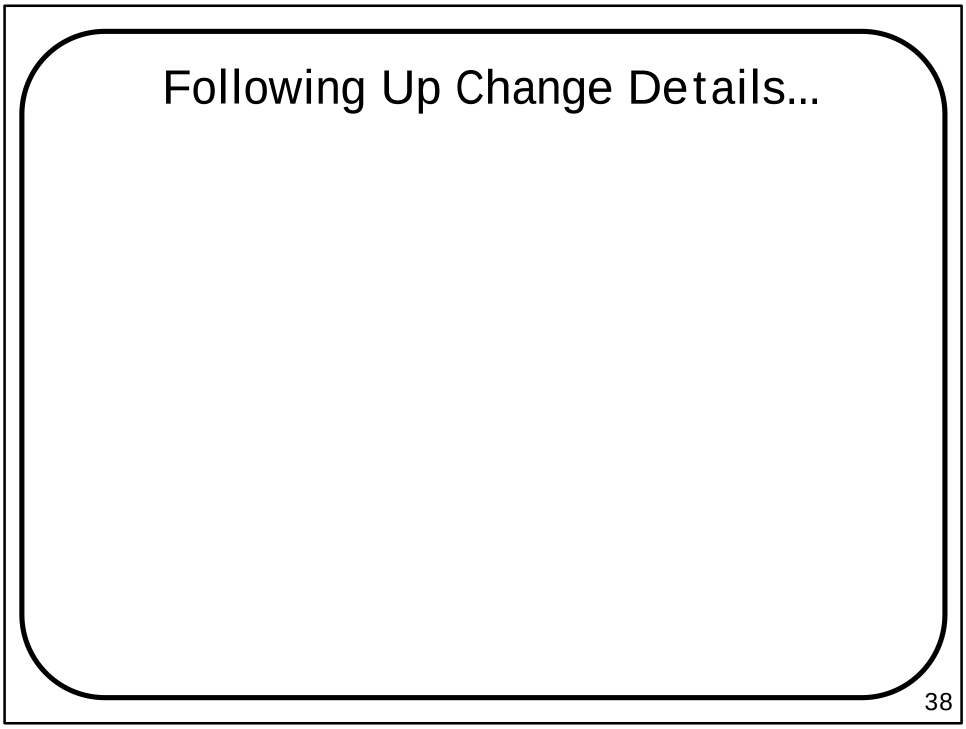
36



About Changes...

Who requested, what instigated, who authorised, etc.

37



Following Up Change Details...

38

Discussion...

Benefits?

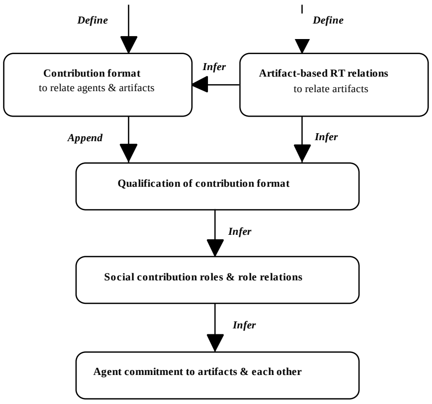
- Deal with:
  - ☞ Info absence
  - ☞ Supplementary info
  - ☞ Human aspects
  - ☞ Continuous change
- Analytic foundations
- Project management
- ...

Issues?

- Automation
- Scaleability
- Firm resistance
- Accountability:
  - ☞ +ve = learn & share
  - ☞ -ve = blame & hide
- Categories
- ...

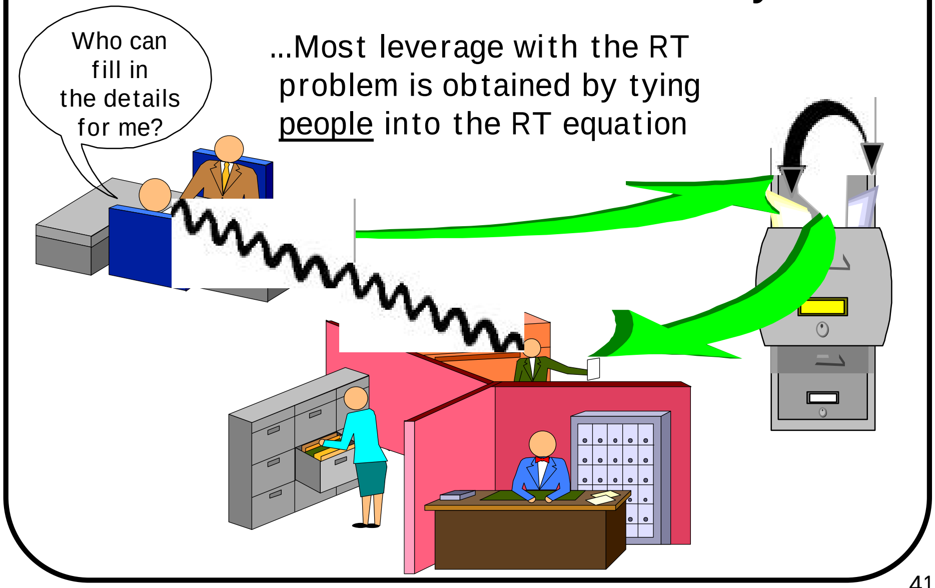
Full Time Recap - A Solution...

- Crux of RT problem is location & access of personnel
- Model social infrastructure underlying reqs artifacts based on concept of “contribution structures”
- Augments artifact-based RT with contribution structures at each step & uses this info for personnel-based RT + much more...


Define
Define
Contribution format
to relate agents & artifacts
Infer
Artifact-based RT relations
to relate artifacts
Append
Infer
Qualification of contribution format
Infer
Social contribution roles & role relations
Infer
Agent commitment to artifacts & each other

The Point To Take Away...

...Most leverage with the RT problem is obtained by tying people into the RT equation



For Further Details...

I can be contacted at:

Department of Computing
Imperial College of Science,
Technology & Medicine
180, Queen's Gate
London
SW7 2BZ

oczg@doc.ic.ac.uk

Papers can be found at: ftp dse.doc.ic.ac.uk
dse-papers/viewpoints