10 Questions* you should answer before you get serious with your research

<NIER: New Ideas and Emerging Results>

Christoph Dorn
www.infosys.tuwien.ac.at/staff/dorn

* Attribution goes to Prof. Richard N. Taylor, UCI
Overview

• Motivation for this talk
  – There are more things to consider before you get serious than you think
  – Applies mainly to writing small/medium/large proposal but also major papers → BUT: not about the actual writing

• 10 Questions:
  – General discussion
  – Eat your own dog food: applied to RiFlexS: Rigorous Flexible Systems

• Feedback
  – Any additional aspects you consider important
  – On RiFlexS (grilling me softly ...)

[Image of a grill with the text: "grilling me softly..."]
10 Questions at a glance

Focus on:
1. Goal
2. Tangible Benefits
3. Technical Difficulties
4. Approach Elements
5. Overcoming Challenges
6. Unique/Critical Output
7. Potential Spin-Off
8. Measuring Progress
9. Current Status
10. Work Schedule
1. Goal

- *What is the main goal of your work?*
  - the ultimate target,
    - not the solution
  - formulated precise and short,
    - not the approach
  - sets the scope
Goal - RiFlexS

• Enable the development of interaction-intensive systems that seamlessly and simultaneously support tightly controlled user actions and flexible ad-hoc interactions.

• Sub-objectives
  – Specification of such systems (interaction aspects)
  – Informing the designer on expected system behavior, trade-offs, and constraints
  – Focused Infrastructure (runtime support)
A Motivating Scenario/Story

• Having a good scenario is important
  – Guides your thoughts
  – Keeps you down to earth
  – Provides scope and boundaries, assumptions
  – One of the earliest “discussion” document

• What is a good scenario
  – Balance between complexity and simplicity
  – Easy to relate to (the more familiar the better)
  – Achievable
  – Realistic assumptions
  – Actual problem

• Better to have two or three complementary scenarios, but at least you should have one!
RiFlexS scenario

- Design a system for monitoring critical infrastructure
  - Guaranteed behavior: ensure all event sources are monitored
  - Flexible behavior: allow operators to dynamically compose sources
2. Tangible Benefits

• What are the tangible benefits to society of achieving that goal (i.e. why should anyone pay for this work)?
  – Why is your research important, why should anyone care?
  – How does solving the problem result in benefit? Why is this a relevant problem?
  – Who are the stakeholders (who uses your output, who benefits indirectly)?
Tangible Benefits - RiFlexS

• Enable novel types of applications
  – Enable flexibility in constraint-driven environments without losing control
  – Enable control in user-driven environments without losing flexibility

• Applicable to example domains:
  – Hospital domain: enhance precisely specified processes with participant flexibility
  – Collective Intelligence domain: collaborative efforts evolve easier through on-demand coordination/control mechanisms
3. Technical Difficulties

- What are the technical problems/challenges that make the goal difficult to achieve (i.e., why hasn’t this been done already)?
  - If it’s a problem, but simple, let industry do it
  - What are the tricky aspect that are most likely preventing you from success
    → risk assessment
  - Not about the effort for implementation or evaluation
  - Know your related work
Technical Difficulties - RiFlexS

- **Fundamental property:** Control and Flexibility are diametric
- **[Design]** Specifying various degrees of control and flexibility, respectively their trade-offs
  - How much flexibility is possible while maintaining a certain minimum level of control/awareness and vice versa
  - How to (dynamically) shift between flexibility and control
- **[Deployment]** Collaboration patterns not designed for composition
  - Meaningful integration of different mechanisms for control and flexibility
  - Enforcing control across pattern boundaries
  - Designing for flexibility without jeopardizing control
- **[Analysis]** Human behavior is inherently fuzzy
  - Realistic assumptions when analyzing composite pattern design
  - Correctly interpreting human behavior at runtime
4. Approach Elements

• *What are the main elements of your approach?*
  – Focus on Methodology, Steps
  – Where to gain requirements from, what to analyze
  – What process to follow (e.g., iterative, exploratory)
  – How to evaluate
  – Not about the output
Approach Elements - RiFlexS

• Explorative and iterative development
  – Investigate different mechanisms for flexibility and control (breadth of patterns)
  – Refinement of mechanisms (depth of patterns)
  – Investigate different coupling intensities (pattern mapping degree)

• Prototyping and evaluation (comparison with solutions based on existing techniques)

• Two application domains:
  – Adding flexibility to control-centric applications in critical domains such as health care or infrastructure monitoring
  – Adding control to flexibility-centric, Internet-scale, collaborative web applications (e.g., collective awareness)
5. Overcoming Challenges

• How does your approach handle the technical problems that have prevented progress in the past (i.e., what makes you think you can do it when no one else could before)?
  – No, the answer is not your intellect and ingenuity (there are most definitely more intelligent people out there)
  – Apply concrete Techniques, Tools, (conceptual) Frameworks, Principles
    → how do these assist
      • Using machine learning techniques, reasoning techniques, formal specification techniques, architecture styles, ...
  – Refer again to related work
Overcoming Challenges - RiFlexS

• Specify precisely the dependency types among collaborators → these can then be managed
  – characterize collaborations in terms of architectural styles
• Specify precisely the user action range, and loci of control
  – Constraints on collab patterns (when to relax, when to enforce)
• Introduce mappings between patterns
  – Investigate which pattern properties can be used as indicator in another pattern, under what assumptions/conditions
  – Remaining within a single style simplifies analysis, but makes specification often awkward, non-intuitive, ...
  – Primarily: mapping between control-flow (i.e. process view) and structure (i.e., architecture component + connector view)
6. Unique/Critical Output

• What are the unique, novel, and/or critical technologies developed in your approach?

  – Types of output: SotA study, model, modeling language, algorithm, (programming) framework, reference design/architecture, design methodology, proof, user study
  – Might want to distinguish according to design-time, deploy-time, run-time, ...
  – Beware: evaluation output is not a contribution per se
    • User study for proving prototype’s usability → no contribution
    • User study for gaining insights into user behavior → contribution
  – Avoid featuritis: it can do, x, y, z, and a and b. → focus on a few main contributions
Unique/Critical Output - RiFlexS

• Pattern composition specification language
  – Properties, influence propagation, constraints (under specific assumptions)
• Techniques for composed pattern analysis
  – Get some assurance that the system will work as intended
  – Resource utilization, response time, agility, failure likelihood
• Proof-Of-Concept Runtime framework for composed pattern execution (monitoring, enforcement, ...)
  – Specific, focused set of collaboration capabilities
  – (grounded in actual interaction technologies such as XMPP only in demo applications)
7. Potential Spin-offs

• *What are the potential spin-offs or other applications of your work?*
  – Improve your (chances of) impact
  – Show that your research is not some obscure, academic exercise
  – Helps to identify additional stakeholders, new perspectives, opportunities for future research (proposals)
Potential Spin-Offs - RiFlexS

• Utilize in extensible interaction frameworks where user can dynamically compose patterns
  – E.g., imagine a wikipage where you could dynamically add a survey to a particular content selection.

• Refine in mechatronics domain for coordinating among all stakeholders
  – Integrated (automated) coordination among customer, product manager, requirements engineer, architect, analysts, electrical engineer, hydraulics engineer, embedded software engineer, tester, safety, documentation, etc.
8. Measure Progress

• *How can progress be measured (i.e., how can anyone tell if/when you’ve succeeded)?*
  – Milestones: specific properties of your output at particular stages (time frame)
  – Evaluation: show that your output has the claimed properties/benefit (intermediate and final)
    • Use case evaluation/demonstration, performance measurements (incl. comparisons), user study, statistical tests, simulation
  – Side Aspect: Enables you to ensure you are doing research correctly.
Measure Progress - RiFlexS

• Iterative Approach (2x) – Milestones for each version and deliverable:
  – Specification Language for Multi Pattern Architecture
  – Analysis Tools
  – Runtime Framework
  – Scenario/Demo app

• 2nd iteration improves on expressiveness, scope, features, stability.
9. Current status

• What have you accomplished so far? What knowledge/previous work are you building upon?
  – Demonstrate familiarity/experience with the topic under investigation
  – Demonstrate that you have reason to believe in your success
  – Demonstrate that you won’t start from scratch but spend resources wisely/not reinventing the wheel: “standing on the shoulders of giants”.
  – Has implications on your work plan.
Current Status - RiFlexS

- First attempt at a *human Architecture Description Language* (hADL)
  - Focuses on flexibility
10. Work plan

• *What is the schedule for your (remaining) work?*
  – Brings together approach, milestones, output
    • apply SW Engineering models (iterative, waterfall, …)
  – Effort estimation for implementation and evaluation
    • Don’t ask: how long will Task A take, rather what can I achieve in 1 month, 3 months …
  – Risk mitigation
  – Research collaboration: when and how to interact, how to split the work
    • e.g., clearly separated research lines for PhD students
  – Keeps you focused
Work plan - RiFlexS

• 3 year project,
  – 6 months warm up (requirements refinement, background, SoTA, technology evaluation, ...)
  – 12 months 1\textsuperscript{st} iteration
  – 12 months 2\textsuperscript{nd} iteration
  – 6 months extended evaluation, writing up

• 2 PhD students
  – Focus on modeling and analysis
  – Focus on arch-2-code mapping and runtime enforcement
Conclusions

• Answering the 10 Questions is not done on a day, or week, or month.
  – Some you already know, for some parts you have to start from scratch.
  – Some more, some less relevant for your particular purpose
• A lot of effort
  – But **start small** and fit all answers on one A4 page.
  – Improve **iteratively** and **discuss**
  – Unfortunately you can never be sure that you done it correctly/completely ➔ live with it.
• Makes you aware what you actually want to do.
  – Not how to write/structure a proposal (very specific aspects for different funding sources)
  – But helps immensely because the core content is mostly there
Thanks for listening

• Questions and Feedback now!
  – 10 Questions
  – RiFlexS
10 Questions/Aspects relations

- **Goal**
  - is relevant because has

- **Tangible Benefits**
  - underlying problem unresolved because of

- **Difficulties**
  - tackled by strategy

- **Approach**
  - appropriate for addressing

- **Overcoming Challenges**
  - using techniques

- **Spin-Off**
  - may also be used for

- **Output**
  - results in

- **Schedule**
  - refines

- **Progress**
  - according to

- **Status**
  - informs

- **TU Wien Distributed Systems Group**