What does it mean to apply a method?
–
Towards a Practice Theory for Software Engineering

Dittrich, Yvonne

"What does it mean to use a method? Towards a practice theory for software engineering." *Inf. and Softw. Technology* 70 (2016): 220-231.

Yvonne Dittrich

Professor in Computer Science
IT University of Copenhagen
Adjunct Professor at the IIT Mandi

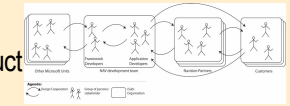
Software Development Beyond the Project – Sustaining Software Ecosystems
©Dittrich 2015

ERP implementation as design: Looking at PD for means to facilitate knowledge integration ©Pries-Heje, Dittrich 2009

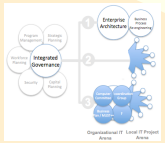
Adapting participatory design to design information system with rural Ethiopian community
©Zewge, Dittrich, Bekele 2015

Satellite image analysis for Irrigation Advice for Himalayan Farmers
https://saiafarm.github.io

Architecture Awareness in Long-Term Software Product Evolution ©Unphon, Dittrich, 2010



Organisational IT managed from the shop floor
©Bolmsten, Dittrich 2016

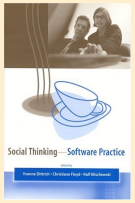


Cooperative Method Development (CMD)

The research approach 'Cooperate Method Development' combines qualitative empirical research with method, tools and process improvement. The research approach has been successfully applied in cooperation with industry

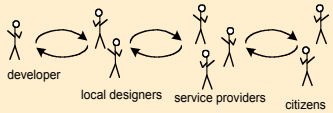
Achievement in numbers:

- 6 companies and organisations engaged in projects using CMD
- 4 externally funded projects applied the approach
- 7 journal articles, 5 book chapters and 8 conference articles.



MIT Press

PD in the Wild
Evolving Practices of Design in Use
©Dittrich et al. 2002



Foregrounding data in co-design–An exploration of how data may become an object of design
©Seidelin, Dittrich, Grönvall 2020

The Role of Ethnographic Studies in SE Research
©Sharp, Dittrich, De Souza. 2016

A conceptual Framework to Study the Role of Communication through Social Software for Coord. in Globally-Distributed Softw. Teams
©Giuffrida Dittrich. 2015

End User Development

Engagement in the development of the End User Development community. I have hosted the International Consortium on End User Development in 2013. My research here bridges from technical implementations to EUD practices and organisational IT management.

Achievement in numbers:

- 8 external partners – 4 software developing companies and
- organisations and 4 user organisations –involved in substantial projects
- 4 externally funded projects addressed related research
- 1 keynote, 5 journal articles, 2 edited conference proceedings,
- 4 book chapters ,9 conference articles.

Cooperative and Human Aspects of SE

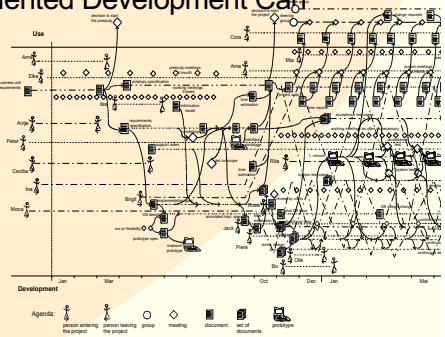
Continuous contribution to the development and shaping of a now established community through the editing of special issues and as workshop co-organiser, PC member and contributor. .

Achievement in numbers:

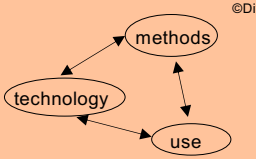
- 8 software engineering companies and organisations involved in projects.
- 4 externally funded research projects
- 19 journal articles, 2 Special Issues and 1 edited volume, published with MIT Press, 1 monograph, 4 book chapters, 20 conference articles.

Learning through Situated Innovation. Why the specific is crucial for Participatory Design Research
©Dittrich, Eriksén, Wessels. 2014

How Use Oriented Development Can Take Place
©Dittrich Lindeberg 2004



What Does it Mean to Use a Method– Towards a Practice Theory for SE
©Dittrich 2016



Use Orientation in SE

Though many researchers address HCI and User Experience, very few do so from a software engineering point of view. I address the usability of the methods for use oriented design and development of software and the cooperation between users and developers in software engineering.

Achievement in numbers:

- 7 external partners – 3 software companies and organisations and 4 user organisations –involved in substantial projects addressing use-orientation in software engineering
- 6 externally funded projects addressed related research
- 1 keynote, 5 journal articles, 2 edited volumes, 1 monograph, 3 book chapters, 13 conference articles.

Roadmap

- “Bad Practice or Bad Methods?”
- Method: Conceptual argument validated by
  empirical findings
- The practice concept in social theory.
  Kant, Wittgenstein and the Pragmatic Turn
- What are Social Practices?
- Software Engineering as Epistemic Practices
- Methods as ‘Practice Patterns’
- What does it mean to use a method?
- Implication for Research, for Practice, for Teaching

‘Bad Practice or Bad Methods’

‘Bad Practice or Bad Methods’

The ‘Bad Practice’ position

- Deviation from the ideal process, resembling a mathematical proof, is due to ‘errors’ and ‘infelicities’.

- Jørgensen, M., & Boehm, B. (2009). Software Development Effort Estimation: Formal Models or Expert Judgment? *IEEE software*, *26*(2), 14-19.

- “We can’t simply conclude that what we observe in projects today is the best way to do something.”
  Parnas in Parnas, D.L., B.Curtis. "Point/counterpoint.” *Software, IEEE* 26.6 (2009): 56-59.

‘Bad Practice or Bad Methods’

The ‘Bad Methods’ position

- Empirical research on Software Engineering shows the influence on situated contingencies on the adaptation and interpretation of methods and principles.
  Button & Sharrock, Rooksby et al. among others.

- ‘The rational Design Process: Why and how to fake it.’ Clements and Parnas.

- The Agile Manifesto

- ‘Good architectural reasons for bad architectural documentation’
  Unphon & Dittrich, ‘Software architecture awareness in long-term software product evolution’, JSS 2010

‘Bad Practice or Bad Methods’

Maybe it is neither nor…

… maybe we just don’t understand what methods are

… and what it means to use them

In lieu of a method section

Philosophical argumentation

- Development of a set of concepts based on related philosophical argumentation
  - Wittgenstein, Schatzki, Knorr Cetina, …
- Relating it to relevant Software Engineering Literature
  - Mathiassen, Floyd, Fitzgerald …
- Supported by empirical evidence
  - Own and others' research

- Criteria:
  - Logically stringent argumentation
  - Empirical support

The practice concept in social theory: Kant, Wittgenstein and the Pragmatic Turn

The practice concept in social theory

K. Schmidt, The Concept of 'Practice': What's the Point? COOP 2014.

Practice is not a new concept (1)

- Plato and Aristotle:
  - Theoria – contemplative activity
  - Poesis – making, producing
  - Praxis – mere activity
- In Enlightenment a new perspective of practice evolved: Practice is something to be improved and therefore subject of systematic collection and research.
  (Which later turns into for rationalization of practice in the Tayloristic sense)
- So also the conceptualization of the relationship between theory and practice changes.

The practice concept in social theory

Kant

"One calls a conceptualization of rules, even of practical rules, a theory when these rules, as principles, are thought of in a certain generality and thus have been abstracted from a multitude of conditions that nonetheless necessarily influence their application. On the other hand, one does not call just any operation a praxis; rather, only such a purposive endeavor is considered a praxis that is taken to be attained by following certain generally accepted principles of procedure."

I. Kant, Über den Gemeinspruch: Das mag in der Theorie richtig sein, taugt aber nicht für die Praxis (Berlinische Monatsschrift, September 1793). In: Werke in zwölf Bänden. Frankfurt a. M.: Suhrkamp Verlag, 1964, vol XI, pp 125–172 Translated and cited by K. Schmidt, The Concept of 'Practice': What's the Point? COOP 2014.

The practice concept in social theory

Wittgenstein's Philosophical Investigation

What gives meaning to language?

- The meaning of words is based on the way we use them in our 'language games'
- The 'rules' in this 'language games' define the meaning a certain term has in a specific context.
- Rules are themselves routed in social practices, they are a commonly acknowledged way of acting.

The practice concept in social theory

Wittgentstein's Philosophical Investigations

"A rule stands there like a signpost.—Does the signpost leave no doubt about the way I have to go? Does it show which direction I am to take when I have passed it, whether along the road or the footpath or crosscountry? But where does it say which way I am to follow it; whether in the direction of its finger or (for example) in the opposite one?"

Wittgenstein, PI 85

"And hence also 'obeying a rule' is a practice"

Wittgenstein, PI 202

The practice concept in social theory.

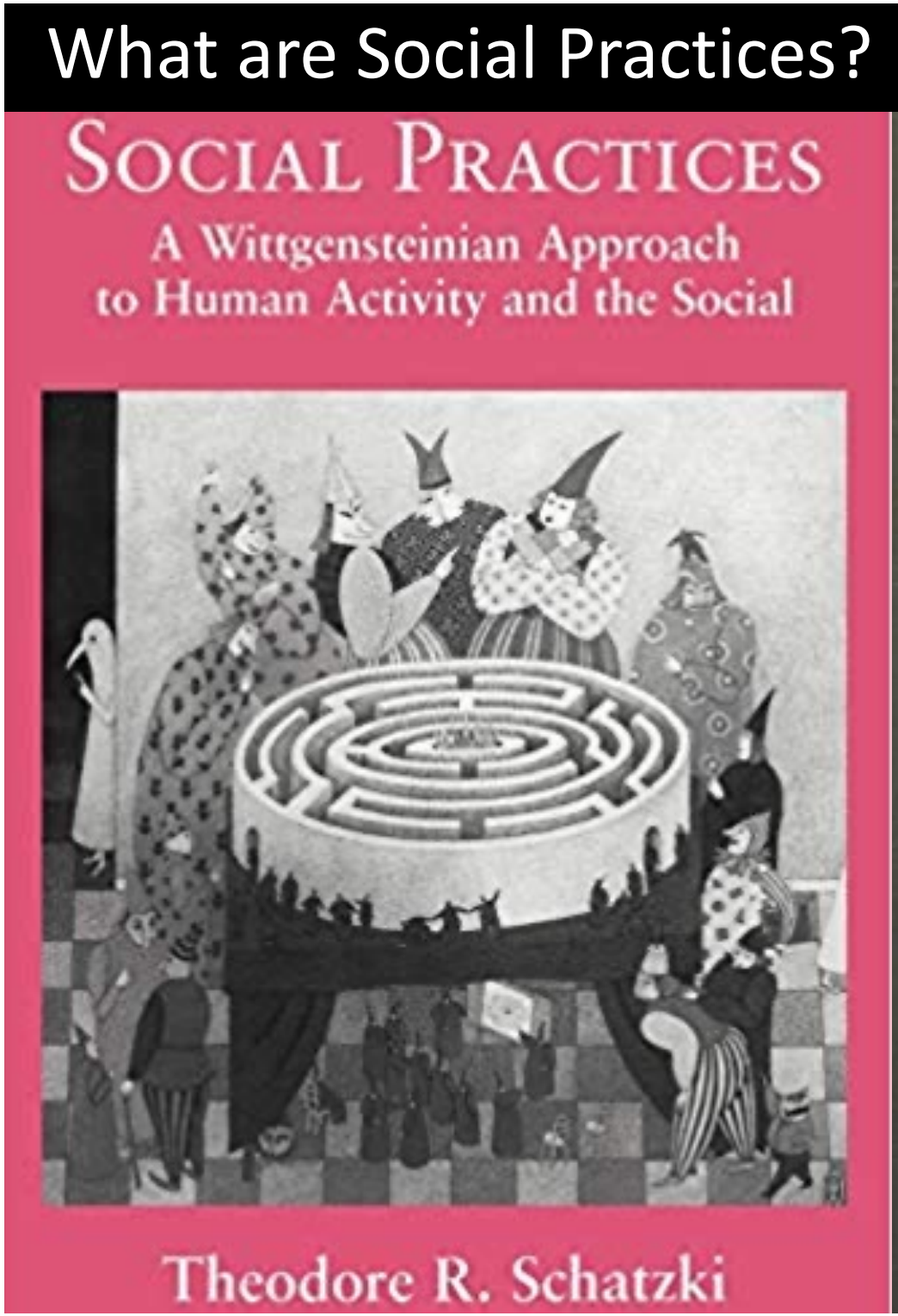
The Practice Turn

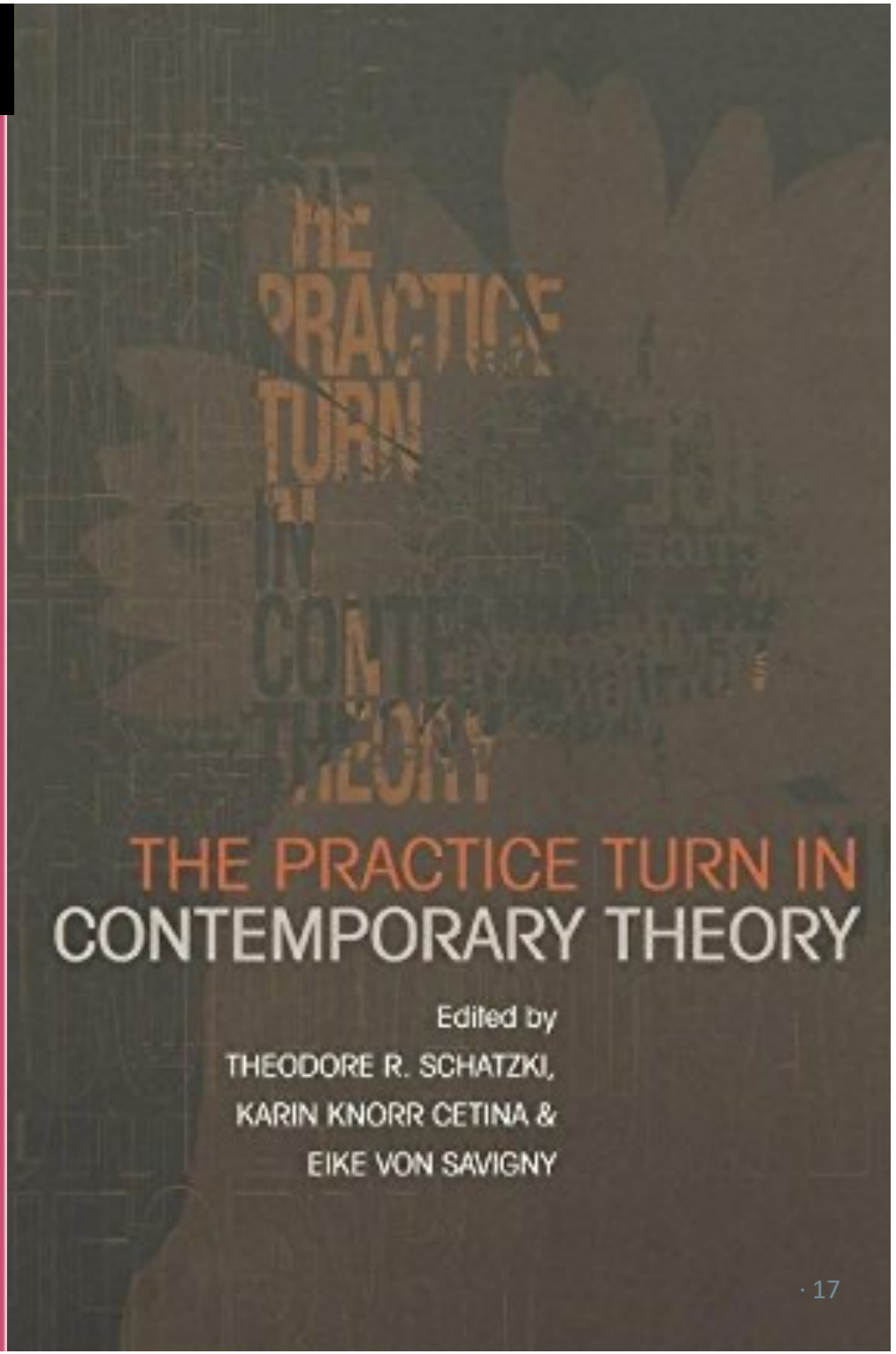
A number of social science approaches refer to Practice as the central concept for understanding social activity.

Schatzki, Knorr Cetina and Savigny publish an Edited Volume on '*The practice turn in contemporary theory*'

What are Social Practices?

What are Social Practices?





What are Social Practices?

Schatzki defines social practices as a

"… temporally unfolding and spatially dispersed nexus of doings and sayings. […] to say that the doings and sayings forming a nexus is to say that they are linked in certain ways.

Three major avenues of linkage are involved:

(1) through understandings, for example, of what to say and do;

(2) through explicit rules, principles, precepts, and instructions; and

(3) through what I will call teleoaffective structures embracing ends, projects, tasks, purposes, beliefs, emotions, and moods."

What are Social Practices?

Two types of practices

Dispersed practices:

widely shared practices, like “asking for and giving explanations, describing, ordering that occur in different sectors of social like.” (Schatzki, p.91)

Integrative practices:

“more complex practices found in and constitutive of particular domains of social life.” (Schatzki, p. 98)

In integrative practices, implicit and explicit rules become more specific and related to the specific ends (teleo-afffective structures) that are constitutive for these practices.

What are Social Practices?

Making sense of each others' actions

- A shared practices helps members to understand each others' action as meaningful with respect to the common goal.
- Practices can be distinguished from ad-hoc behavior. Ad-hoc behavior is sometimes needed to bring things back to normal. (example from Rönkkö, Randall and Dittrich. "When Plans do not work out" CSCW Journal 2005)

What are Social Practices?

What about tools, environments

"Not only people, but **objects (and events) as well acquire meaning within practices**. This occurs, most importantly, whenever objects are used in the performance of constituent actions. […]
Like understanding generally, the understanding of equipment is expressed not only in doings (i.e., uses) but also in sayings. People give names to equipment and say of them that they have such and such practical meanings […] ." (Schatzki, p. 113)

"An integrative practice, consequently, carries interwoven understandings of interrelated equipment." (Schatzki, p. 114)

What are Social Practices?

Practices Change

"I note that among the acceptable actions (and live condition orders) constitutive of a practice's teleoaffective structures are some that have not yet been carried out (or instantiated). Practices found possible novelty in that ***people happen upon new ways of proceeding***, and others deem these ways acceptable, on the background of their participation in practices and familiarity with teleoaffective structures. The understandings that ***organize an integrative practices likewise, though more weakly, open ranges of acceptable doings and sayings broader than the behavior already performed in the practice***." (p. 102)

What are Social Practices?

Adaptation to local contingencies.

Additional Schatzki's concept of integrated practices, we can observe local 'specializations':

- Software engineering differs in Eriksson or at Microsoft.
  (And that can be a source for mis-coordination in distributed software development)
- Each project seems to have its own mix of methods and tools

What are the sources for these contingencies?

Software Engineering as Epistemic Practices

Software Engineering as Epistemic Practices

Knorr Cetina's concept of Epistemic Practices

- Based on her empirical work on development of a particle accelerator.
- Epistemic practices are practices whose objective is bringing something into being that is not yet there and not yet fully known.

'[O]bjects of knowledge in many fields have material instantiations, but they must simultaneously be conceived of as unfolding structures of absences: as things that continually 'explode' and 'mutate' into something else, and that are as much defined by what they are not (but will, at some point have become) than by what they are.' (p. 182)

- 'Knowledge-centred work shifts back and forth between performance of 'packaged' routine procedures and differentiated [epistemic] practices'

Software Engineering as Epistemic Practices

Software Engineering and epistemic practices

- Software engineering is based on established ways of doing design and development guided by
  - explicit and implicit understandings of how things are done
  - explicit rules that are codified in form of methods and techniques
  - teleoaffective structure in form of an idea about a piece of software that is not yet existing.
- Software engineering is geared at unfolding what is to be designed and developed (but not yet there).
- **While its object, the software under development, unfolds, software engineering practices unfold responding to the developing understanding of what is their goal.**

Software Engineering as Epistemic Practices

SE Practices are consciously maintained and adapted

Sigfridsson (2010) “The purposeful adaptation of practice: an empirical study of distributed software development”

Draxler et al. (2014) discuss how teams organize the keeping up to date with developments of their tools and assign members to take on the tailoring and adaptation of new features for the whole project.

Giuffrida & Dittrich (2014) show how a student project team when initiating the project and in occasions of a breakdown discuss and introduce new ways of cooperating and coordinating.

Methods as ‘Practice Patterns’

Methods as 'Practice Patterns'

Webster's:

1: a **regular systematic plan** for or way of doing something 2a: **orderly arrangement** b: **habitual regularity** an orderliness [Latin *methodus* from Greek *methodos*, from *meta* + *hodos* "way]

The Free Dictionary (http://www.thefreedictionary.com/method)

1. A means or manner of procedure, especially a **regular and systematic way of accomplishing something**: *a simple method for making a pie crust; mediation as a method of solving disputes.* See Usage Note at methodology. 2. **Orderly arrangement of parts or steps to accomplish an end**: *random efforts that lack method*. 3. The **procedures and techniques** characteristic of a particular discipline or field of knowledge: *This field course gives an overview of archaeological method*.

Likewise the **SWEBOK** defines methods as with the following sentence:

"Software engineering methods provide an **organized and systematic approach** to developing software for a target computer."

Methods as ‘Practice Patterns’

Mathiassen proposes the following characteristics as defining a method:

An **area of application**: a type of software to be developed using it, a way to organize the development process;

A **perspective** consisting of assumptions about the nature of the system, organizations, the surrounding society and the purpose of the local organization;

Principles for organizing the development process, splitting it into partial tasks;

Techniques of work used in the partial tasks;

Tools used in the application of the technique (diagrams, notations, or computer support).

Methods as ‘Practice Patterns’

Based on empirical comparison of analysis and design methods, Floyd et al. propose to extend this list with three additional characteristics:

Theories: mathematical theories on which e.g. the notations are based, and the underlying theory about what is software development;

Coherence describes whether a method applies a connected set of techniques and tools or whether they are difficult to relate;

Coverage: Which tasks of the development process are supported.

Methods as ‘Practice Patterns’

Relating the SE understanding of methods to Schatzki’s concept of Social Practices

Methods can be seen as complex related sets of

- explicit formulated understandings (notations, modeling languages, concepts, area of application, coverage, the mathematical side of the theories),
- explicit rules (processes, task descriptions, techniques etc),
- and explicit teleo-affective structures (principles, perspectives, theories in the sense of what software engineering is about).
- They might be supported by tools.

These sets of explicit understandings, rules, and teleoaffective structures need to be embedded in concrete pratices.

Methods as ‘Practice Patterns’

But what is the role of methods in SE practice

Software engineers don’t refer to methods in their day-to-day work. Mathiassen et al.

Formal methods methods vs Methods-in-action. Fitzgerald et al.

And also:

Plans and Situated action. Lucy Suchman

Maps and scripts. Kjeld Schmidt

When Plans do not work out. Rönkkö, Dittrich, Randall

Methods as 'Practice Patterns'

But why do we develop and use methods?

"What does man think for? What use is it? – Why does he make boilers according to *calculations* and not leave the thickness of their walls to chance? After all it is only a fact of experience that boilers do not explode so often if made according to these calculations.
But just as having once been burnt he would do anything rather than put his hand into a fire, so he would do anything rather than not calculate for a boiler."

Wittgenstein, PI 466

Methods as ‘Practice Patterns’

Methods can be understood as Practice Patterns

‘A pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over without ever doing the same thing twice.’

Christopher Alexander

- Applying a method or a practice pattern does not mean to do exactly the same as the neighbor project applying the same method.
- The participants in the practice of software engineering act with heed to their knowledge of this complex and related set of (tool supported) understandings, rules and teleoaffective structures.

What does it mean to use a method?

What does it mean to use a method?

Methods – in – Use

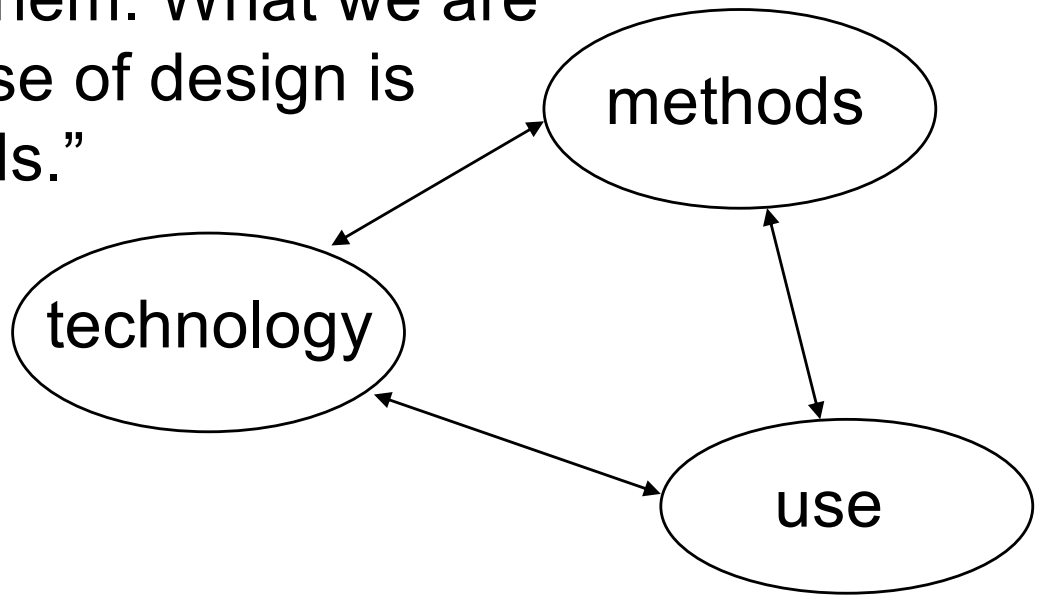
- Methods, understood as practice patterns, are formulations of understandings, rules, and teleoaffective structures responding to known problems maybe supported by tools.
- As any other formulations or expressions, they do not have a meaning per se.
- The participants in a practice make sense of them, given the already established practice.

Again:
"And hence also 'obeying a rule' is a practice"

Wittgenstein, PI 202

What does it mean to use a method?

"We do not apply predefined methods, but construct them to suit the situation in hand. There is no such things as methods per se – what we are invariably concerned with are processes of situative method development and application. We select methods and adapt them. What we are ultimately doing in the course of design is developing our own methods."



© Floyd, C. Software Development as Reality Construction. In: Floyd et al. Software Development and Reality Construction. Springer 1992,

What does it mean to use a method?

The adoption and adaptation of a method is part of the maintenance and changes to Software Development practices referred to above.

What does it mean to use a method?

Adoption and adaptation of methods as meta-work

Articulation work (Strauss 1985)

- planning
- **standardization of cooperation procedures**, (in the following called meta-work)
- articulation of tasks in order for others to relate to it,
- And handling of contingencies that lead to exceptions from the standard

The adoption and adaptation of methods presented in Sigfridsson's PhD thesis and Draxler et al.'s article fall under the concept of meta-work.

What does it mean to use a method?

The observation of meta work practices can be used to distinguish between generation of method use and conscious adoption and adaptation of methods

Dittrich, Yvonne, et al. "Exploring the evolution of software practices." Proceedings of the ESEC/FSE 2020.

https://youtu.be/QBB_z7261ug

How do methods come about?

How do methods come about

Empirically, we can identify two main ways in which methods are introduced:

(1) as abstractions of existing practices, in order to communicate useful practice patterns to newbies and fellow practitioners; and

(2) as output of special practices called research.

Implications

Implications

For Research

- Methods that are research results need to be tested in practice.
- We need to learn from practice:
  - About the contingencies in which software development takes place, so we can device the right flexibility for our methods.
  - About changes to practice that might change the applicability of our methods.
  - About the problems, practitioners face.

Implications

For Practice

Control the methods!
Don't let the methods control you!

- Include the maintenance and change of practices as part of the development (e.g. retrospectives in Scrum)
- Explore new tools and methods and their possible contribution to the current project.
  - Investigate how a method understood as practice pattern can address current problems
  - Investigate how the adoption of the method changes the practices of the team.
  - Consciously adapt practices to the situation at hand.

Implications

For Teaching

- We need to teach methods not as truths but as practice patterns.
- We need to teach how to adopt and adapt methods in a reflective manner.

References

For full references, please refer to

Dittrich, Yvonne
"What does it mean to use a method? Towards a practice theory for software engineering." *Inf. and Softw. Technology* 70 (2016): 220-231.

Dittrich, Yvonne, et al. "Exploring the evolution of software practices." Proceedings of the ESEC/FSE 2020.