Laboratory for Usage-centered Software Engineering

Lab-USE

Interacção 2006

From Activity Theory to Design Practice

Modeling the Activity Context

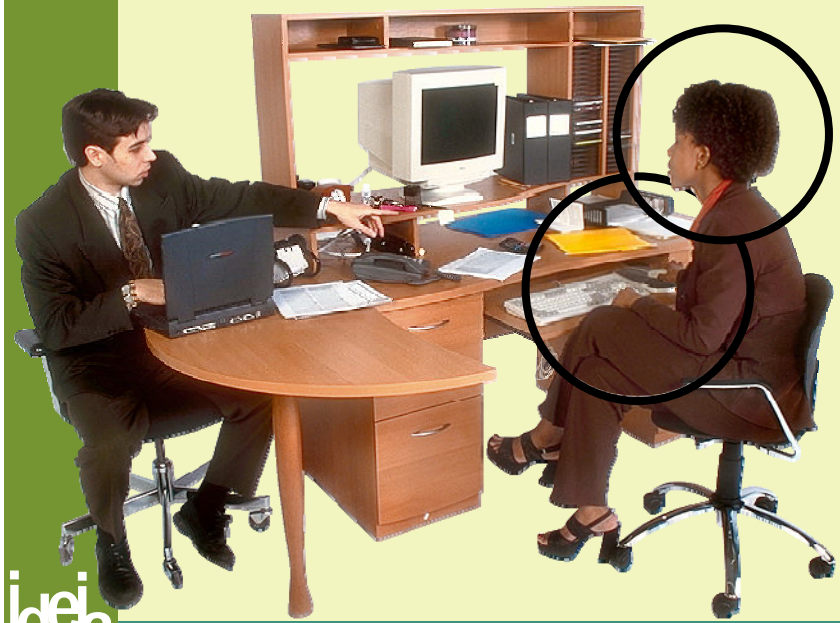
Larry Constantine, IDSA

centro de Investigação e Desenvolvimento em Engenharia Informática e Aplicações

UNIVERSIDADE da MADEIRA

Users or Uses?

- What do interaction designers need to understand to design effective solutions? What do they focus on?
  - users?
  - uses or user performance?
  - context?
- Usage-centered design is a proven model-driven design approach focused primarily on user performance.
- Emerged from invention of essential use cases in 1993.



- Purely pragmatic, heuristic.
- Used successfully in varied projects ranging to over 1000 person-years.
- Simplified abstract models:
  - user roles (not personas)
  - task cases (not scenarios)
  - abstract prototypes

Why Focus on Use, Not Users?

- Users are people, people are complicated.
  - emotional, psychological, social, cultural
  - background, personal history, experience
  - involved in many activities in various contexts
- Product use is only one small element of life.

- Compared to people, interactive use of products is relatively simple.
  - narrow, limited channel
  - specific tasks and activities
  - selected behaviors
  - defined work/social context

The (relatively) simple relationship of users to products is most important for good interaction design.

Why Models?

Model: a simplified abstraction representing selected features and characteristics of other objects.

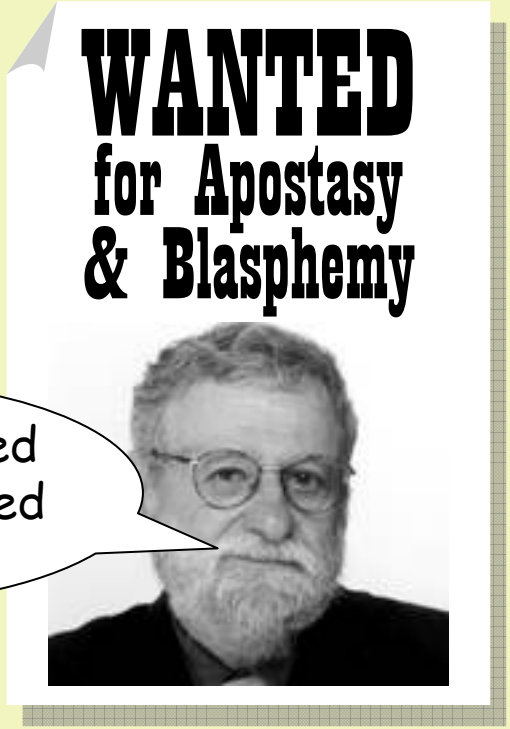
- Building models is easier than building the real thing.
- Models capture, carry, and organize understanding about a problem or possible solution.
- Models permit exploration of the problem and solution space.
- Models can be validated against objective criteria.
- Models can be tested and evaluated.

Model-driven processes:

- provide an “audit trail” of assumptions, of how understanding evolves, and of how solutions are based on these.
- facilitate tracing results back to requirements.
- enable smooth derivation of reasoned solutions.

Why Activity Theory?

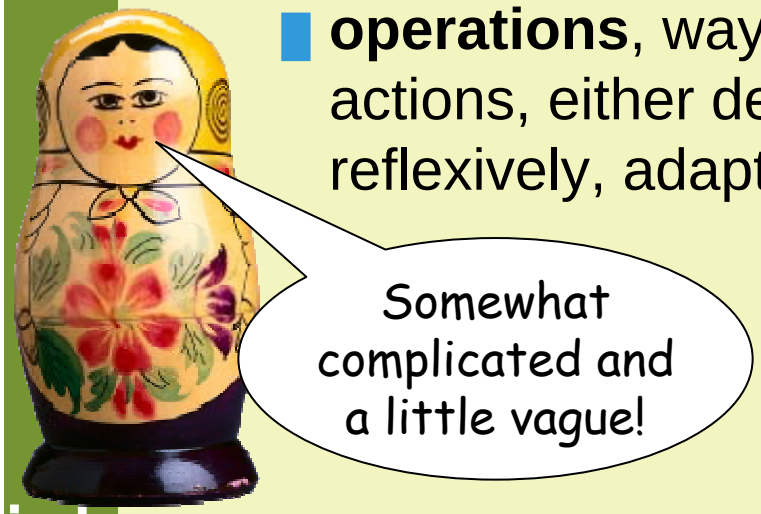
- Don Norman is a trouble-maker. (So am I!)
- Software engineering models like UML largely ignore contextual aspects of user requirements.
- Even in usage-centered design, contextual aspects are loosely defined, weakly structured as "operational profiles."
- Usage-centered models of work capture discrete tasks (essential use cases) but lack straight-forward representation of higher level work abstraction or workflow organization.
- Business process models, scenarios, complex use cases, or compositions of task cases can model workflow through discrete tasks, but
  - often too complex.
  - overly specific, constrained.
  - omit many important contextual aspects.



Activity Theory Condensed

- Created by early 20th century Russian psychologists Rubinshtein, Leontiev, and Vygotsky.
- Popularized by Nardi and others.*
- Not so much a theory as a conceptual framework.
- Some prior attempts to systematize and operationalize.**
- Hierarchical structure of activity (three levels of analysis):
  - **activities** are motivated, purposive, and consist of
  - **actions** directed toward a distinct, specific conscious goal, consisting of
  - **operations**, ways of executing actions, either deliberately or reflexively, adapted to conditions

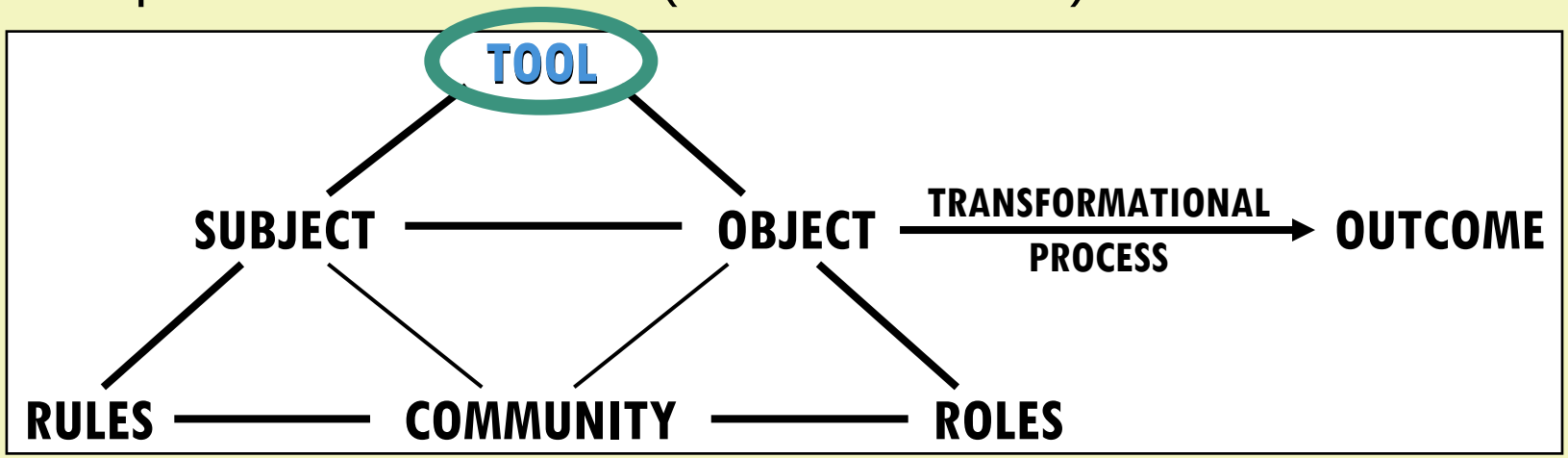
ACTIVITY	–	PURPOSE
↓ ↑		
ACTION	–	GOAL
↓ ↑		
OPERATION	–	CONDITIONS



* Nardi (ed.) *Context and Consciousness.* 1996.
Gay & Hembrooke. *Activity-Centered Design.* 2004.
** Duignan, Noble, & Biddle, 2006
Kaptalinin, Nardi, & Macaulay, 1999

Activity Theory Condensed

- Activity* is performed by a human **agent** (subject) motivated by **purpose** (object or motive) and mediated by **tools** (artifacts) in a transformational process yielding a **result** (outcome) through collaboration with others (**community**) constrained by cultural factors (**rules**) and differentiated responsibilities or **roles** (division of labor).



- All human activity is mediated by **tools**.
- Supporting human activity requires designing effective **tools**.
- The design of effective tools requires insight into **activity**.

* after Engeström, 1999

Interaction in Context

- Work, for example telephone customer support, takes place in a context, such as a call center.
- User tasks are performed within the context of larger activities, both related and unrelated.
- Different activity contexts impact users and how they perform using tools and artifacts differently.
- Analysis models need to reflect understanding of the activities and the context in which they are performed.

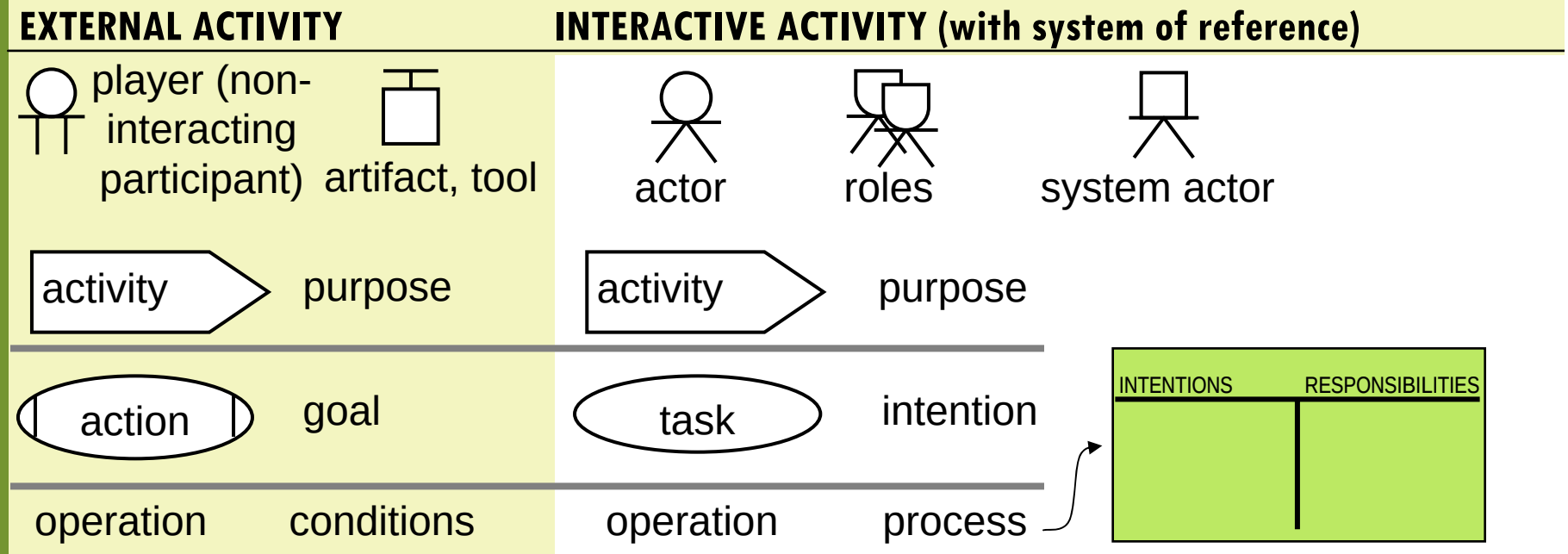
- Applications need to support the activities in which users are engaged within the context in which they are performed.
- Use cases and other models of discrete tasks are not enough!

The Plan

- Formalize, or at least systematize, the modeling of activities.
- Connect task modeling based on essential use cases (task cases) to activity theory via activity modeling.
- Create a single, coherent set of concepts with practical notation:
  - transparent vocabulary and clear, simple concepts
  - simple, easily grasped, memorable notation
  - concise even if not completely precise
- Provide usage-centered design with a well-defined, theoretically sound anchor to the context of work.
  - generalized abstract alternative to scenarios for understanding larger structure of interaction
- Capture and succinctly represent salient information most relevant for interaction design.
- A practical design aid, not a research tool or comprehensive framework for research.

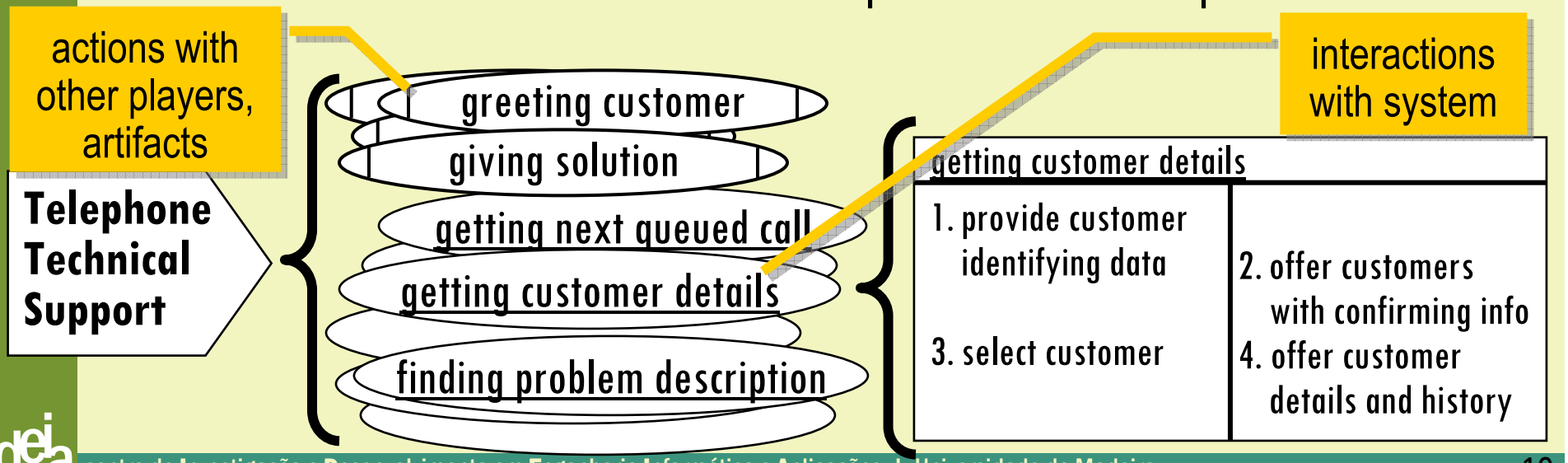
Modeling Interactive Activity

- In practice, must model, connect, and distinguish activities that include user-non-user and user-system interaction:
  - interacting and non-interacting participants
  - relationships among participants, artifacts, and systems
  - relationships among activities and interactive tasks and among external activities and actions



Activities, Tasks, and Operations

- Activities are the larger context of interactive use.
- Activities, Actions/Tasks, and Operations are three levels of analysis for understanding work.
- Activities involve Tasks (interactions with system) and Actions (with other people, systems, and artifacts).
- Activities are unstructured or loosely structured aggregations of Actions and Tasks, helping us identify and understand work in context.
- Tasks consist of one or more Operations or steps.

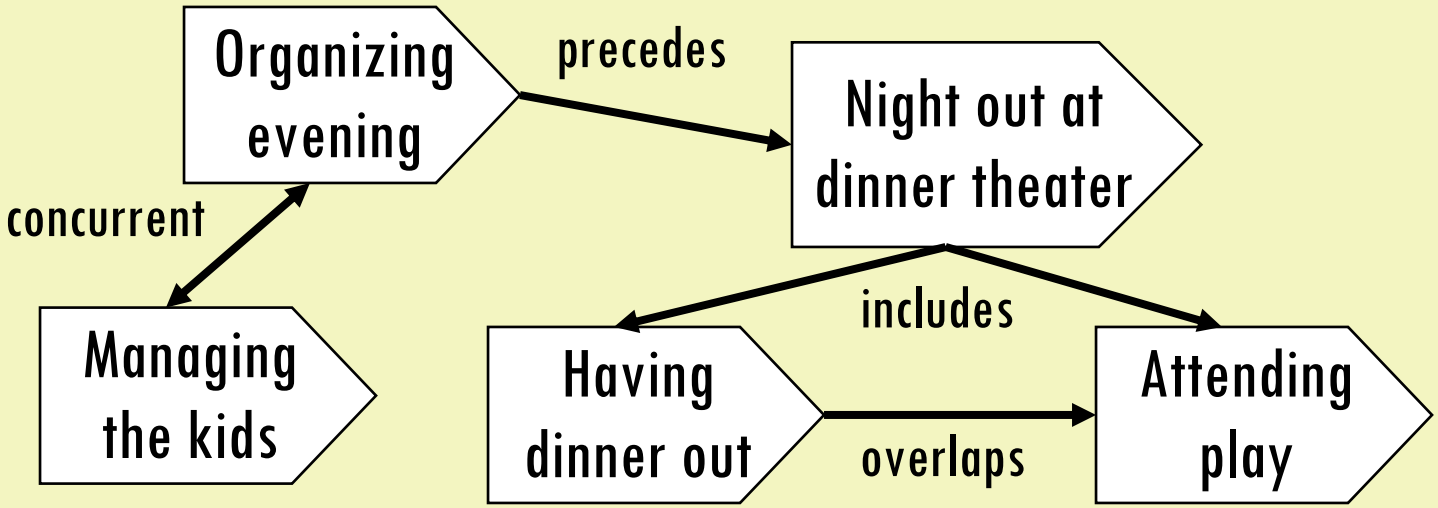


Activities, Tasks, and Operations

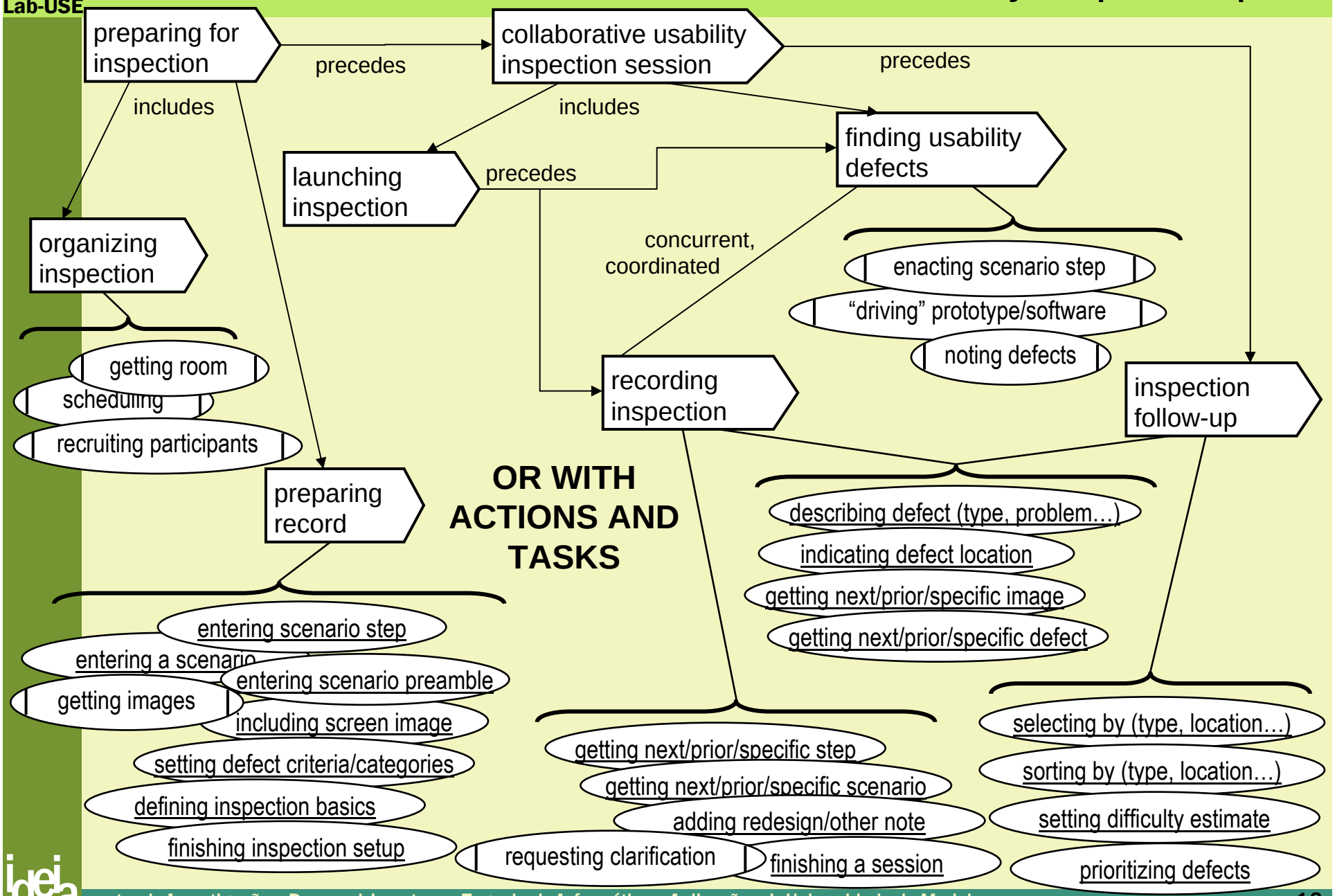
- **Activity**: coherent collection of interrelated Tasks (and Actions) undertaken by Actors (and Players) within some situation for some common purpose.
- **Task**: single, discrete intention within an Activity, consisting of Operation(s); complete, well-defined, and meaningful to an Actor in some Role.
- **Operation**: conditional step of Task performed by Actor.
  - **using public information kiosk**
    - **finding an ethnic restaurant**
      - **selecting ethnic food type(s)**
  - **commissioning robot welder**
    - **validating program download**
      - **starting code comparison**
  - **developing PowerPoint presentation**
    - **animating clipart entrance**
      - **drawing entry path of object**

Interrelated Activities

- Activities can be related in various ways -
  - contains (includes)
  - coordinated (synchronized)
  - concurrent (synchronous, asynchronous, interleaved)
  - consecutive (precedes, overlaps)
  - competes (involves common participants or resources)
  - impinges (in same “field,” affecting an activity)
  - …

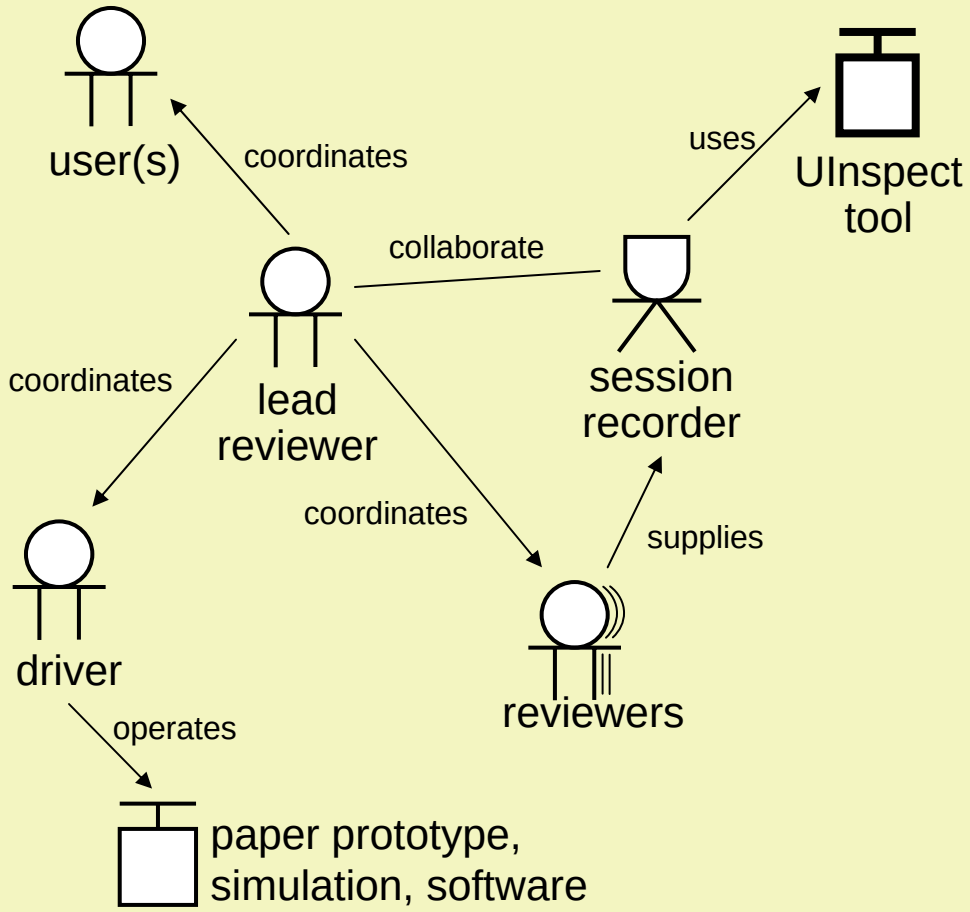


Activity Map Example



Relationships Among Participants

- Actors and players can interact in various ways with each other and with tools/artifacts as part of activities:
  - collaborates
  - competes
  - coordinates/guides
  - supplies/delivers
  - consumes/receives
  - exchanges
  - communicates
  - …



Activity Detail Described

Purpose - motive, objectives, what's it all about

Place and Time - where, under what conditions? physical environment, social context; duration, schedule, frequency

Participation - who's involved? actors (and roles played), players (non-actors), system actors; community of practice; responsibilities (division of labor) and relationships (among participants); tools, artifacts, information sources, other resources used

Performance - characteristics, style; coordination or other relationships with other activities; formal and informal rules of performance

Product - implications for presentation and interaction design of product

Activity Description Illustrated

Recording Inspection

Purpose - quickly and accurately describe, classify, and prioritize identified usability defects

Place and Time - in dedicated room with moderate number of others; moderately formal social setting; focused, time limited (1-3 hours typical)

Participation - recorder, lead reviewer, user(s), observers (uncommon), “driver,” reviewer(s), which may include designers and developers from this and/or other projects; system or design being inspected; possibly reminders (cards, posters) of rules and responsibilities

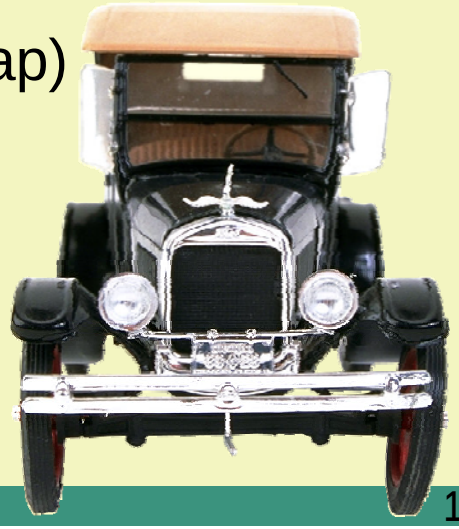
Performance - intense, pressured (up to 100 defects per hour); high volume, moderately complex information from multiple sources in rapid bursts, quick decision making and judgment required; may often need to leave incomplete or backtrack to complete; governed by formal, written rules, assigned roles

The Activity Model

- **Activity Catalog** - Inventory of activities with categorized descriptions (purpose, place, time, performance, participants, tools, rules,…)
  - optionally including inventory of actions/tasks
- **Activity Map** - model of relationships among activities (inclusion, coordination,…)
  - optionally with actions/tasks and relationships

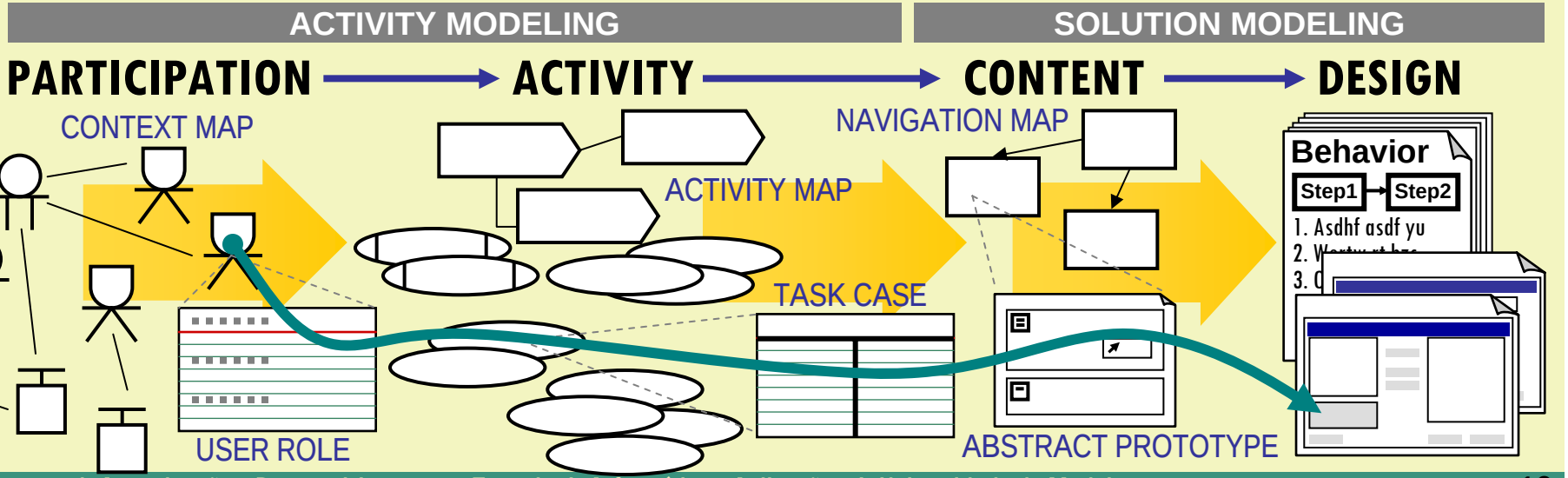
elaboration of task model

- **Participation Map** - models relationships among participants (players, actors, roles) and artifacts
  - system-centered (extension of Context Map) in relation to system of reference and to other tools/artifacts
  - activity-centered - models relationships among participants in relation to activities



Model-Driven Process Overview

- Users are modeled as **roles** played by **actors** in **activities** with **players** (other participants) and **artifacts** (tools) plus **system actors**.
- **Activity** is modeled as **actions** and **tasks** (essential use cases) composed of **operations** in process narrative (intentions, responsibilities).
- Organization and functional contents of user interface are modeled by **navigation map** and canonical **abstract prototypes**.
- **Visual and interaction design** derives from abstract prototypes.
- Models drive the entire process. Design elements trace directly to content supporting tasks needed to perform roles within activities.



Activity-Centered Design Problems

- Activity context for general purpose tools may be highly variable, difficult to analyze, and impossible to anticipate.
- One approach in such circumstances is context-sensitive interfaces like Office 2007 “tool ribbons” and “context tabs.”
- What if you and the software guess wrong?

Post-modern logic for interaction design…

“Anything can be in more than one place at any time.”

- Redundant presence and multiple paths generally increase the probability of user success.
- Interfaces adaptable by users can fit unanticipated activities.
- But, there are so many user roles in so many activities…

Activity Modeling Potential

- Highlights and clarifies relationships among collections of tasks (and actions) without excess precision/constraints.
- Models aggregation of task cases (essential use cases) into larger, more loosely or variably defined collections.
- Highlights relationships among user actors and other players and with other artifacts.
- Organizes contextual aspects known to be important in guiding visual and interaction design.

Activity-centered interface architecture -

- WYNIWYG: tools and materials needed for performance of an activity consolidated into a common region of interfaces.

  (Architectures based on category hierarchies often inefficient.)
- Helps to clarify system boundary decisions:
  - actions can become tasks (or tasks actions).
  - external artifacts can move inside system of reference.

Activity Modeling Challenges

- New models, another notation to learn. (But quite simple.)
- Not supported by modeling tools. (At least not yet.)
- Need to clarify definitions and separation of concerns for roles and activities. Consider:
  - Roles - Casual entertainment browser, Targeted ticket seeker, Business-motivated investigator
  - Activities - Casual entertainment browsing, Targeted ticket seeking, Business-motivated investigation
- Need guidelines, templates, fully worked-out examples.
- Need supporting software tools for both software engineering and interaction design.
- Plus, ultimately, incorporation into

  Old **M**ired **G**roup standards.

Selected Resources

Brown, R. B. K., Hyland, P., & Piper, I. C. "Eliciting and Specifying Requirements for Highly Interactive Systems Using Activity Theory." *Interact 2005 Proceedings - IFIP WG 2.7 / 13.4 - User Interface Engineering.*

Constantine, L. L. (2004) "Beyond User-Centered Design and User Experience." *Cutter IT Journal, 17,* 2: 2-11.

Duignan, M., Noble, J., & Biddle, R. (2006) "Activity theory for design." *Proceedings, HWID 2006.* University of Madeira.

Engeström, Y., Miettinen, R. & Punamäki, R-L. (Eds.) (1999). *Perspectives on Activity Theory.* Cambridge University Press.

Gay, G. & Hembrooke, H. *Activity-Centered Design.* (2004) MIT Press.

Kaptalinin, V., Nardi, B. A., & Macaulay, C. (1999) "The Activity Checklist." *Interactions 6,* 4: 27-39

Nardi, B. (ed.) (1996) *Context and Consciousness.* MIT Press.

Norman, D. (2005) "Human-Centered Design Considered Harmful." *Interactions,12,* 4: 14-19; also at jnd.com

Acknowledgement - For generous support and feedback from my colleagues at LabUSE: Nuno Nunes, Leonel Nóbrega, and Pedro Campos.