

Task I

- Sites we evaluate:
- Oikotie <http://www.oikotie.fi/>: Elisa Rakkolainen, Joonas Pajunen, Pertti Huuskonen, Mikko Majander
- New York times reader: <http://timesreader.nytimes.com/timesreader/index.html?campaignId=34W8F>
- KP-Lab search 2d.mobile.evtek.fi/shared-space
- But before you start evaluating you have to describe the users, social and physical environment and users tool ecology...

Users

- You need to describe users' characteristics, their segments, needs, wants, habits, preferred usage trends, context (social and physical) and tool ecology
- How?



Personas, Profiles, Actors, & Roles

Modeling users to target
successful product design

Jeff Patton

jpatton@acm.org

AgileProductDesign.com



the future of software development



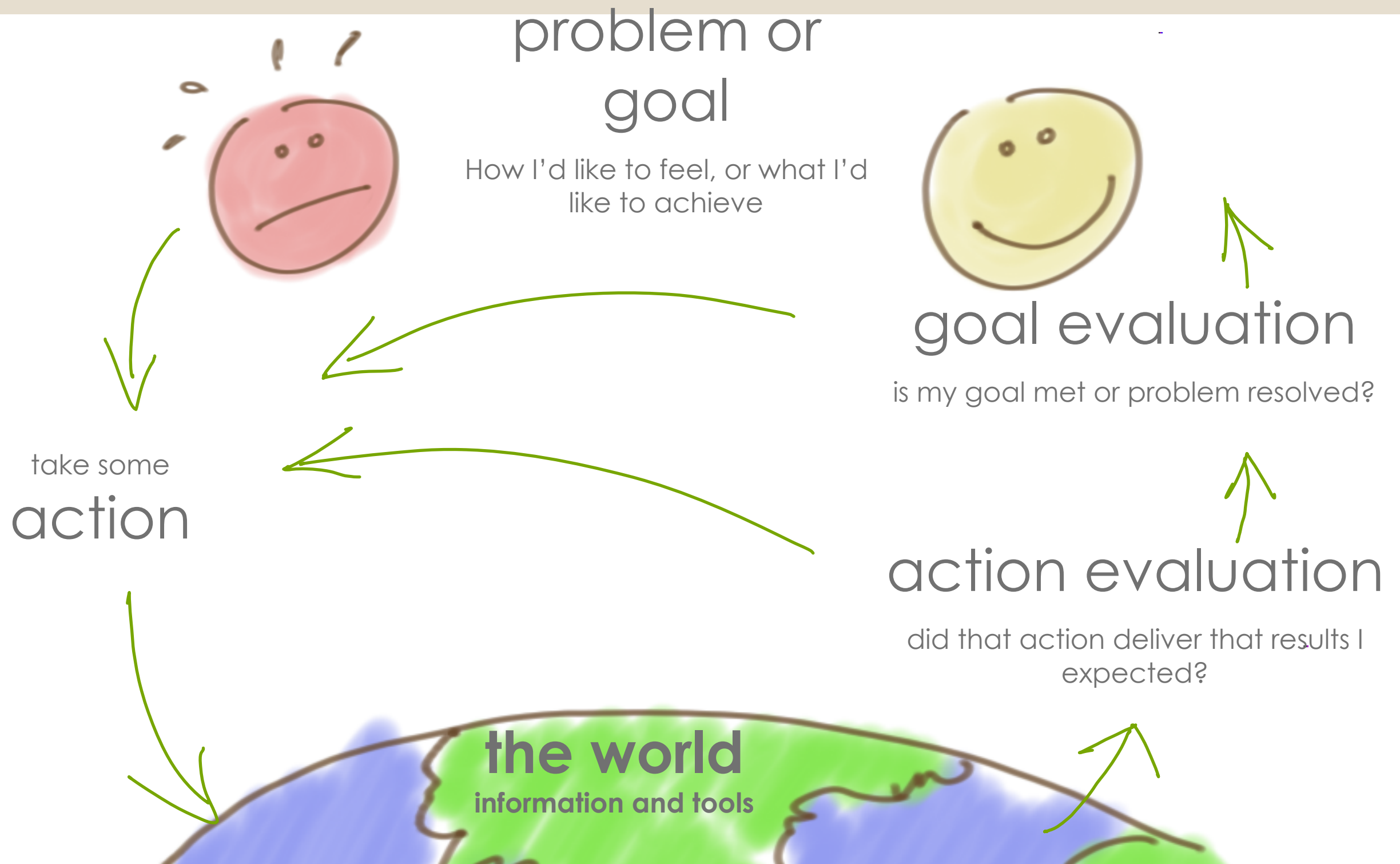
Today we'll cover these three areas

Understand:

- The user model's place in a software development process
- How to build simple relevant user models
- How to leverage a user model to make design decisions

I hope to demystify what often seems a confusing subject in software design and development

 Norman's simple model for a human in pursuit of a goal





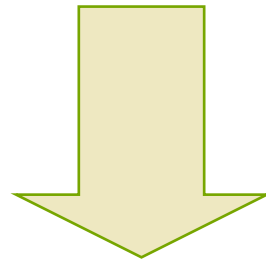
Distilling this down to goals, tasks, and tools



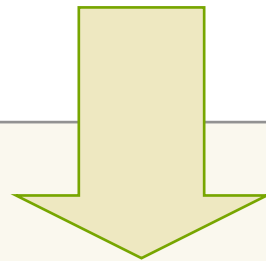


Software contains features that support a number of tasks and a number of goals
Software products support a variety of users and their goals.

goals



tasks



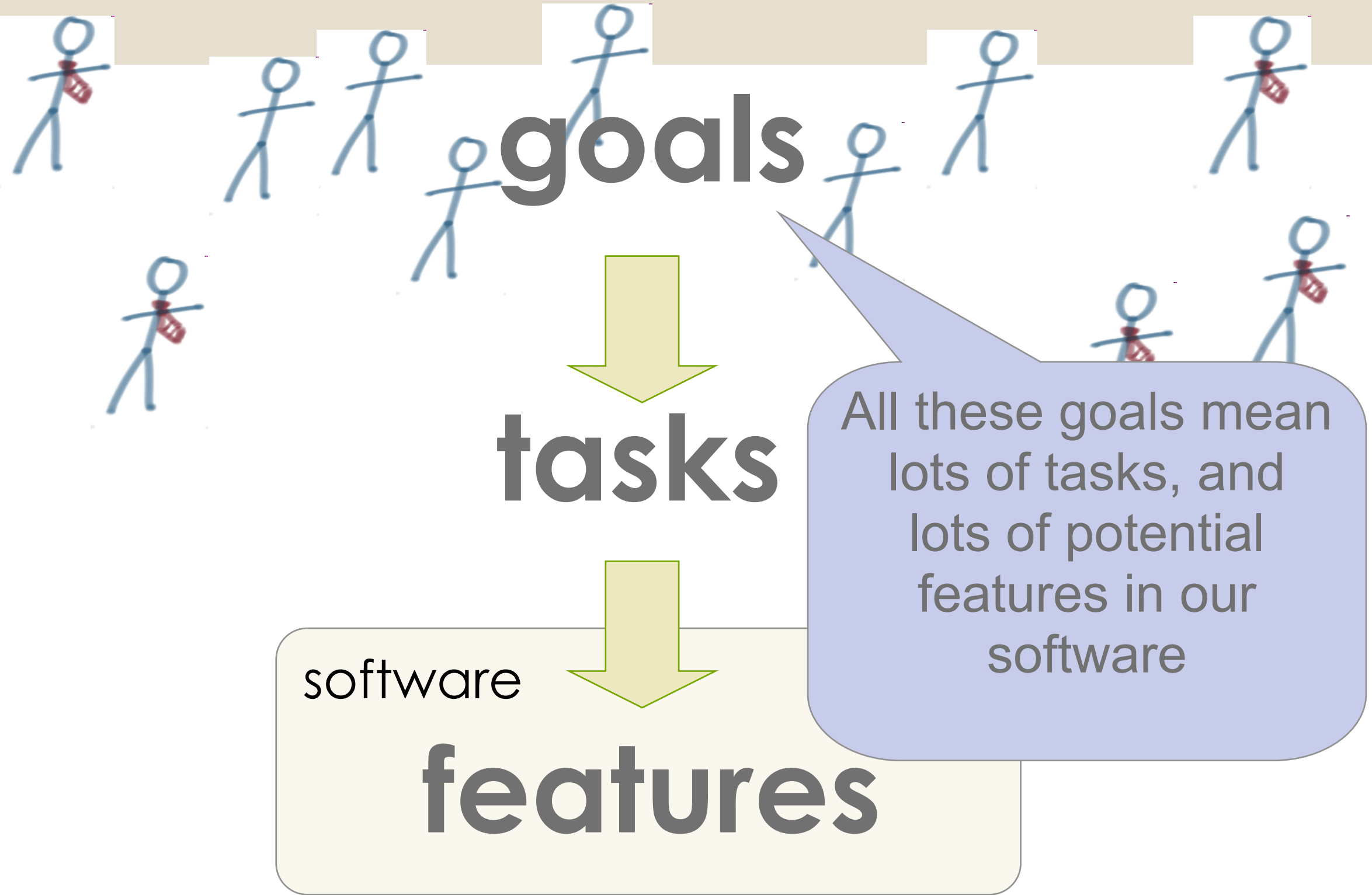
software

tools

features



In organizations where users are paid to use the software, user goals are driven by business goals





Having a good list of users helps us understand functional scope

How many different types of users will use this software?

What goals will they be in pursuit of?

What tasks will they need to perform?

Which of those tasks will the software we design support?



Look closer at the people engaged in using your tool – what about them has relevance to the tool's design?



What do your users know about using computers? - assuming we're building software

What do they know about the goal they're attempting reach? Have they done this before?

How often do they do this?

When and where are they when they'll use the software you design?

If they use other software like this – what expectations might they have about your software?

Questions like these help us understand characteristics our software should have to best serve these users

How do we go about
describing users in the
most relevant way?



The humble “actor” gives a common name for a user type



In use case modeling, actors are people who interact with our system.

They're often described using job titles or a common name for the type of user.

- accounts payable clerk
- manager
- cashier
- customer



The “role” names a relationship between a user type and a process or a software tool



A user role general refers to a user’s responsibility when using a piece of software or participating in a business process.

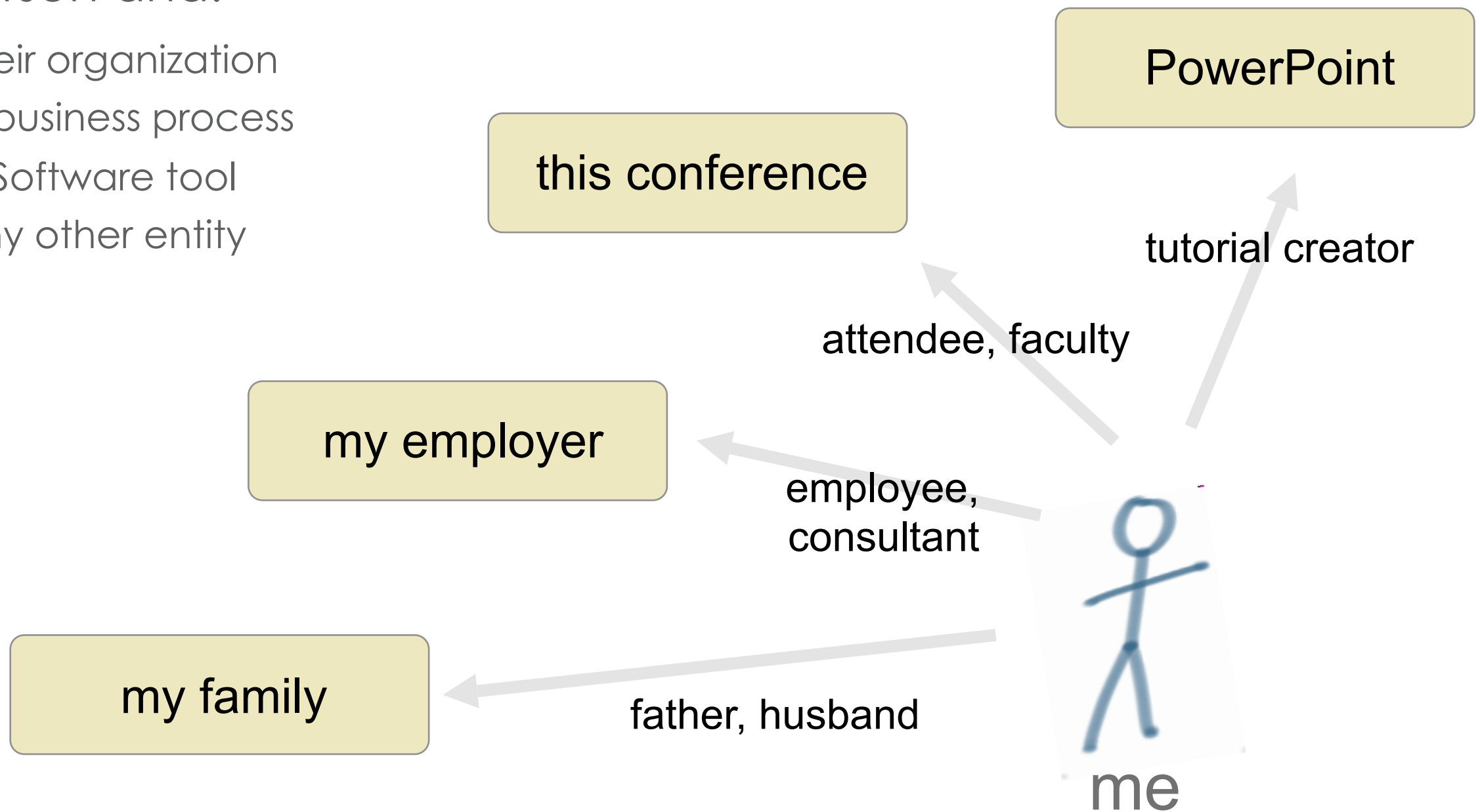
- AP voucher enterer
- administrator
- on-line payment checker



Both actors and roles name a relationship with some entity

That relationship may be between a person and:

- Their organization
- A business process
- A Software tool
- Any other entity





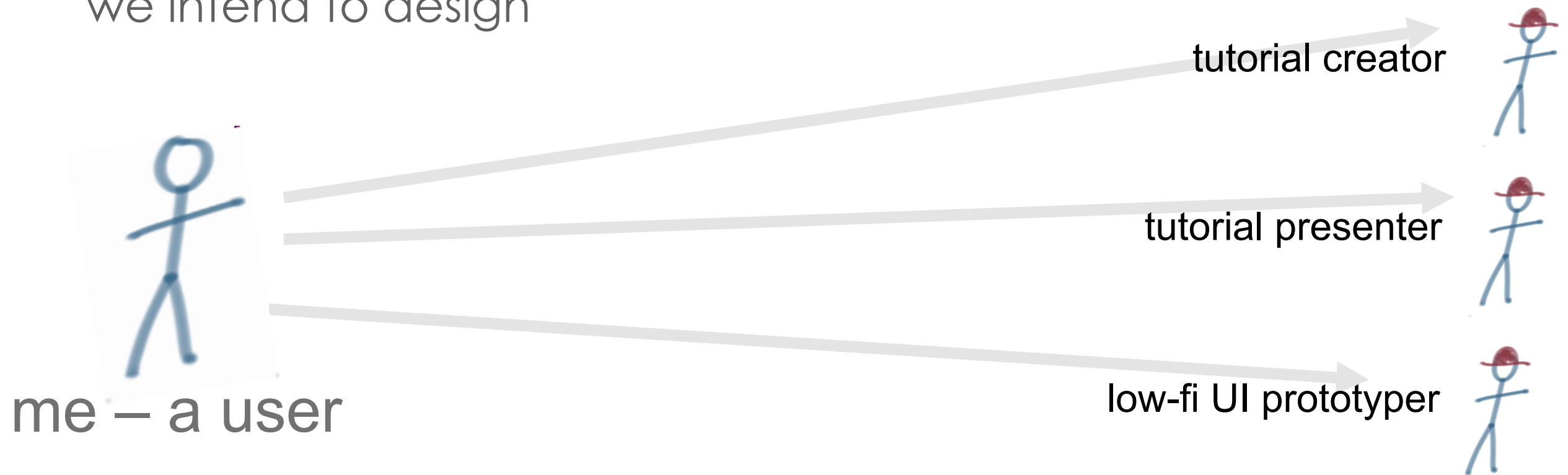
Both actors and roles name a relationship with some entity

An individual may change their role as their goal or responsibility changes.

Changing roles is like changing hats

For our purposes, that entity is the software we intend to design

PowerPoint





enough talk:

Let's practice thinking about roles



You have the site to evaluate so, brainstorm with someone (i.e., form pairs, if you want to work alone ask someone to help you for this task).



xxxx?

In a small group – 3-4 people, brainstorm a list of candidate roles for the system



xxx?

Try using the form “thing-doer” such as “website browser” or “apartment renter”



xxx?

Remember the primary rules of brainstorming:

- No discussion – just ideas
- Quantity matters more than quality
- Keep it fun – suggest silly roles

Timebox this to 5 minutes



Prioritizing user types is important

For the software tool we intend to build, which user types are the most relevant to the design?

This depends on the business case.

Why is the software product being built/exists?

What business objectives do we hope to achieve?

Which of these user types is it most critical that we support to achieve our objectives.

Refer to these users types as primary, or focal

For a typical system, expect 2 or 3 focal users



Choose focal users – the users that best advance SDs business objectives

With your group, choose 2 or 3 focal roles for the SD's new website.



Profile users to identify relevant characteristics about them

To help us understand the characteristics of our users that might have bearing on our design, construct a profile containing information about the type of user relevant to the software being created.

1. **# of users** that occupy this user type
2. **General responsibilities or activities**
3. **Computer skills**
4. **Domain expertise**
5. **Goals:** how does this software tool help this user reach their goals?
6. **Pain Points:** what nagging problems can this software help solve?
7. **Usage Contexts:** where will this software be used?
8. **Software Ecosystem:** what other software tools does this user type rely on?
9. **Collaborators:** who does this user work with to help reach their goals?
10. **Frequency of Use:** how often is this type of user likely to use this software?
11. **WRITE THIS DOWN INTO THE WIKI:**





Creating profiles from assumptions and facts

Quickly creating profiles from assumptions allows us to find out what we do and don't know about our users.

There's danger in basing critical decisions on software functionality on assumptions. But, before allocating time to research, the assumption based profile will help you estimate how much research you'll need.

Interaction designer that create personas from assumptions refer them as and **assumption-based persona**, or a **persona hypothesis**



Backfill profiles with facts

Given assumption based profiles, you can identify the areas where your information is sparse or incomplete. You can use research to backfill your profiles with facts in critical areas.

- Interviewing users from target user groups **(Use this)**
- Observing users
- Questionnaires
- Existing published demographics
- Existing published research **(Use this)**
- Customer service records and representatives
- Sales and marketing
- Usability testing
- Focus groups
- **Use the (use this) to do some research on the gaps you have**



Interview someone nearby

Interview technique: ask your interview subject to recall a specific event and describe to the best of their recollection how that event took place.

Ask them to describe their experience when e.g. they tried to find an apartment for rent...

- Where were they?
- How long did it take?
- What computer equipment or software was used?
- What did they most enjoy about the experience?
- What was most annoying about the experience?

1. WRITE THIS DOWN INTO THE WIKI, + include back up from existing research:



Distill your user model to communicate information most relevant to the design of the software

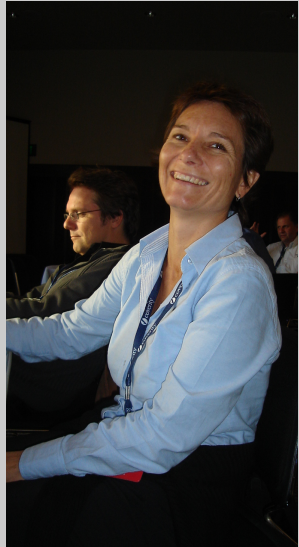
Of the assumptions and facts gathered, what are most relevant to the design of this software?

What could you remove to more concisely communicate to:

- analysts
- UI designers
- developers
- business stakeholders



Personas make user data more tangible



Jutta

Frequent Conference Speaker

“I really appreciate efficient conference organizers – the ones that value my time.”

Jutta has an over-stuffed conference schedule speaking at over a dozen conferences a year internationally. She travels to US conferences from Germany where she lives. She has one published book and is working on her second. Speaking at conferences allows her to share her ideas with others, promote her work, and network with colleagues to share information and experience.

Over the years Jutta has learned the idiosyncrasies of various conference presenters. - some are more efficient than others. She appreciate those that are early with reminders for due dates and forthcoming with information she needs to put together submissions. She's on some conference website every month – but they all vary a bit and it's frustrating to find the critical information she needs on a particular website she only sees once every few months.

She's been using computers since she was young, and although much of her

Profiles contain general characteristics about your groups of users.

A persona is an archetypal user that is derived from specific profile data to create a representative user.

A persona is more tangible, less ambiguous, easier to envision, easier to empathize with.

Personas with irrelevant or stereotypical information in them will damage user understanding and empathy.



Characteristics of a good persona

Name

A role or job title

Quotes in the personas language

Relevant demographics

Descriptions that reveals goals, motivations, pain points

Descriptions that describe primary activities this user type will engage in.



Build a simple persona from your profile data

Include:

- Name
- A role or job title
- Quotes in the personas language
- Relevant demographics
- Descriptions that reveals goals, motivations, pain points
- Descriptions that describe primary activities this user type will engage in, where and in which social relations




1. WRITE THIS DOWN INTO THE WIKI:



Feature opportunities describe the good ideas the good ideas that result from thinking about your users

As you discuss, speak with, and observe your users, you'll get great ideas for product features – features that will really help your users.

Include these feature opportunities in your profile or persona



Design imperatives describe good characteristics the software should have based on the user type

Inside your user profile are clues about the type of user interface and user interface characteristics needed by your user.

Document these as design imperatives.

Think about:

- **ease of learning**
- **retention** of learning
- **efficiency** of interaction
- **reliability** of interaction
- user **satisfaction**
- user **convenience**
- necessity for **proficiency**
- importance of **accuracy**



Discuss and record feature opportunities and design imperatives

What feature opportunities are particularly valuable to this user type?

What characteristics must the design have to be suitable for this user type? (design imperatives)

- ease of learning
- retention of learning
- efficiency of interaction
- reliability of interaction
- user satisfaction
- user convenience
- necessity for proficiency
- importance of accuracy

1. WRITE THIS DOWN INTO THE WIKI:





User modeling distilled

1. Identify actors or roles - take your pick, or mix as you see fit
2. Prioritize based on relevance to the product's business case
3. Profile to identify details relevant to design
4. Personify to better communicate user types
5. Identify feature opportunities
6. Identify design imperatives
7. Communicate your user model with their relevant feature opportunities and design imperatives – this communicates the relevance of your user model



User models are often not leveraged

Leverage your user models

- Prioritize them to help with adding or removing functionality from scope
- Identify user test subjects
- Identify alpha/beta testers
- Compare them with your eventual actual users to identify bad assumptions and new user constituencies
- Post them in the area your team works to help team members empathize with target users and make better tactical design decisions

Understanding users is
critical to getting value
out of our software.

Modeling users is a simple first step to clearly communicating our design target to everyone involved in software design and development.