

### Motivation

- Current *state-of-the-practice* in the software industry is:
  - **Bloated software.** Lots of features for all potential users – *confusion*.
  - **One-size-fits-all.** Users usually adapt to software, while we expect the opposite.
- This applies to many different categories of software:
  - **Personal Software**  
Productivity applications, email clients, etc.
  - **Enterprise Software**  
Financials, HR, Customer Relationship Management, Supply Chain Management, etc.
- This software *cannot be finely customized*, which may render the software *useless* for certain categories of users.

### Most Affected Users

- People with cognitive, sensory, and motor impairments
- Elderly people
- Children
- Novices
- For these people fine-grained customization that suits their abilities is *critical*.

### Our Approach

#### Goals (what the users want)

✓ Gather user requirements. Represent these requirements as **goals**. Create a **goal model** that represents all the **alternatives** for achieving these goals. Each alternative assigns certain **tasks** to the user.

#### Preferences (how the users want their goals achieved)

✓ Represent **preferences** as **softgoals** (goals that require solutions that are “good enough”). Determine the positive/negative **effect** of each alternative on these softgoals and use it to prioritize among the alternatives.

#### Skills (how their abilities constrain the way their goals can be achieved)

✓ Identify **skills** that are required to carry out tasks that are needed to fulfill user goals. Disallow alternatives that assign tasks to users who don't have the necessary skills. E.g. The task *Dictating* requires *Voice Production & Spoken Language* skills.

### Example



### Generating Customized Software

- Associate **generic software architecture** to the goal model.
- Map the *selected alternative* to the corresponding *customized software architecture*.
- Deliver the customized system.

### Case Study

- **Communication system for people with cognitive impairments** (conducted together with the University of Oregon)
  - Goal models with more than 350 goals and 400 tasks.
  - Number of alternatives reached  $10^{10}$ !
  - We are working on creating a generic architecture for the case study.

### Future Work

- **Extend the approach beyond personal software**
- **Runtime Customization**
  - Monitor for changes in user skills and preferences and adapt the software accordingly by selecting the *new best alternative*.
  - This can be done by:
    - Humans – **Adaptability**.
    - The system itself – **Adaptivity**.