AI Methods for Smart Environments

A Case Study on Team Assistance in Smart Meeting Rooms

Thomas Kirste

thomas.kirste@uni-rostock.de

www.informatik.uni-rostock.de/mmis

Background:
Dynamic Device Ensembles

The Challenge, in a Nutshell

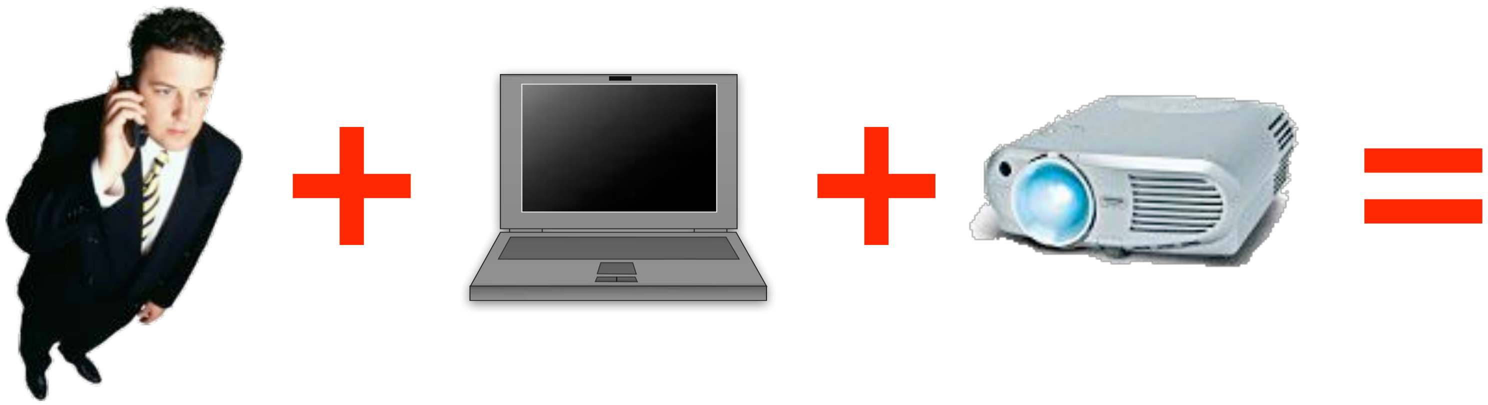
+

+

=

- You walk into the conference room,
  you connect your notebook to the projector,
  and what you get is: …

The Challenge, in a Nutshell



- … and what you really wanted, was this:
- … why don‘t you get this?
  - The devices in your environment are not able to **assist.**
  - The devices in your environment do not support **spontaneous cooperation.**
- **Needed: techniques for the (1) <u>spontaneous</u> (2) <u>cooperative</u> (3) <u>assistance</u> of devices**

The UbiComp Vision

- “digital intelligence” everywhere
- Why? – Invisible and unobtrusive support of everyday activities: **Ubiquitous Assistance**

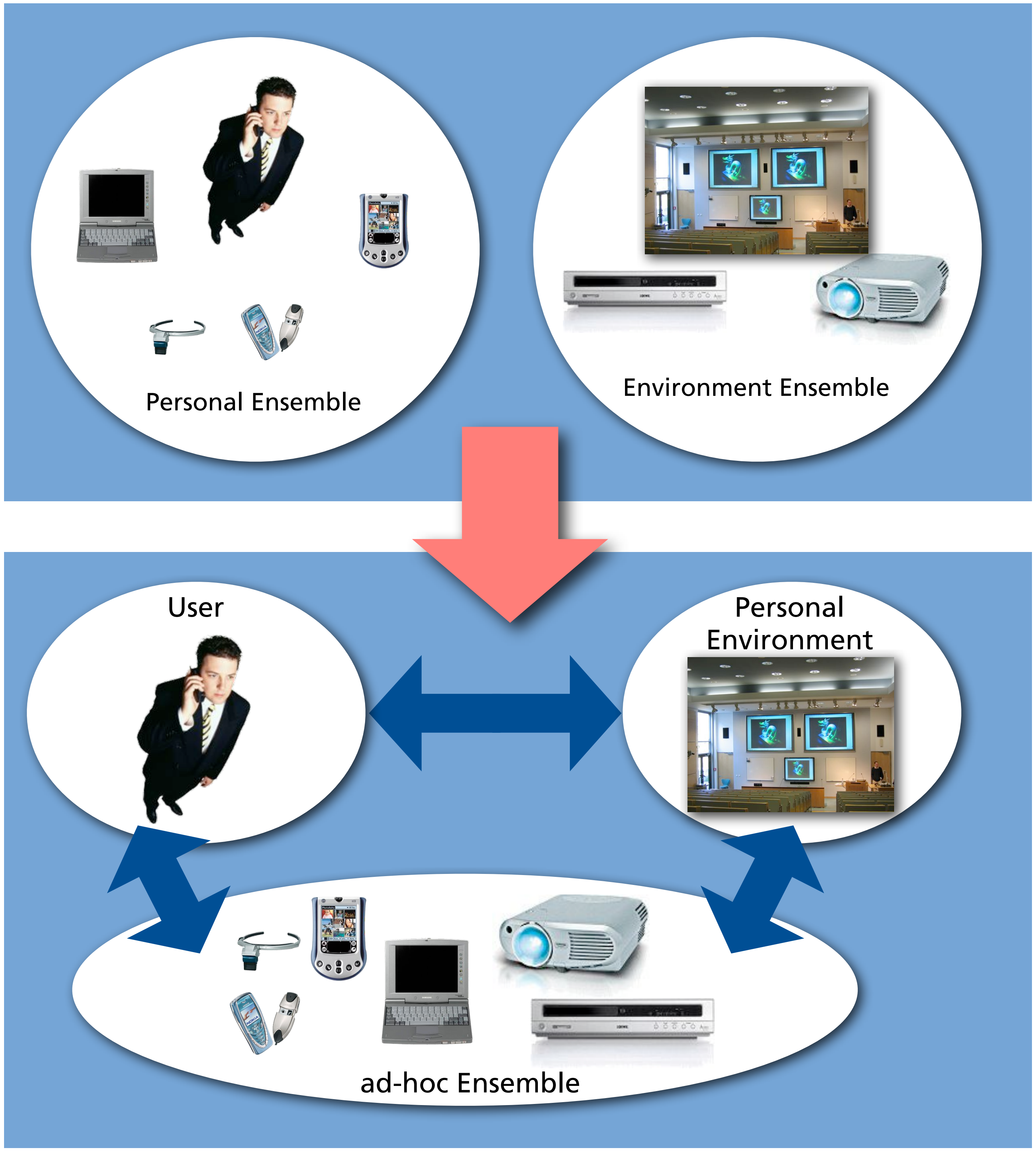
Challenge?

- How do you interact with invisible devices?
- How do you use functionality you‘re not aware of? …
- … and that is distributed across hundreds of devices?

„Ubiquitous Computing enhances computer use by making computers **available throughout** the physical environment, while making them **effectively invisible** to the user“
(M. Weiser)

(c) Philips

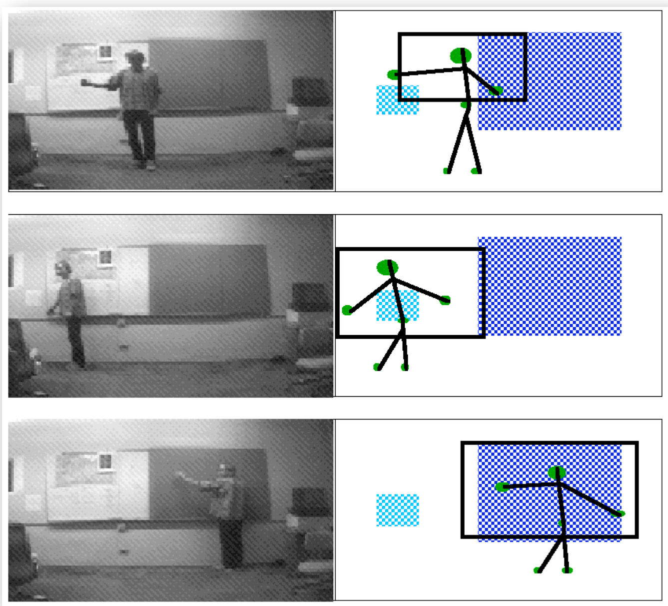
The Ensemble Challenge



- How can the ensemble find out, *what* the user *would* want, if he knew, what he *could* want?
- And how then would the ensemble determine, *how* achieve this *what*?

- How to answer these two questions, without requiring **global / centralized knowledge** about the user and the capabilities of the ensemble?

NWU Intelligent Classroom



- Classroom knows the speakers strategies for presenting things
- Classroom knows, which physical events (location changes, pointing gestures, keywords) accompany a state transition in the speaker's presentation process
- The classroom is able to infer the speaker's current presentation goal
- The classroom performs actions (e.g., zooming in) that support this goal without requiring explicit interaction.
  - [Franklin & Hammond, Northwestern University, 2001]

    www.cs.northwestern.edu/~franklin/iClassroom/index.html

EMBASSI Smart Living Room

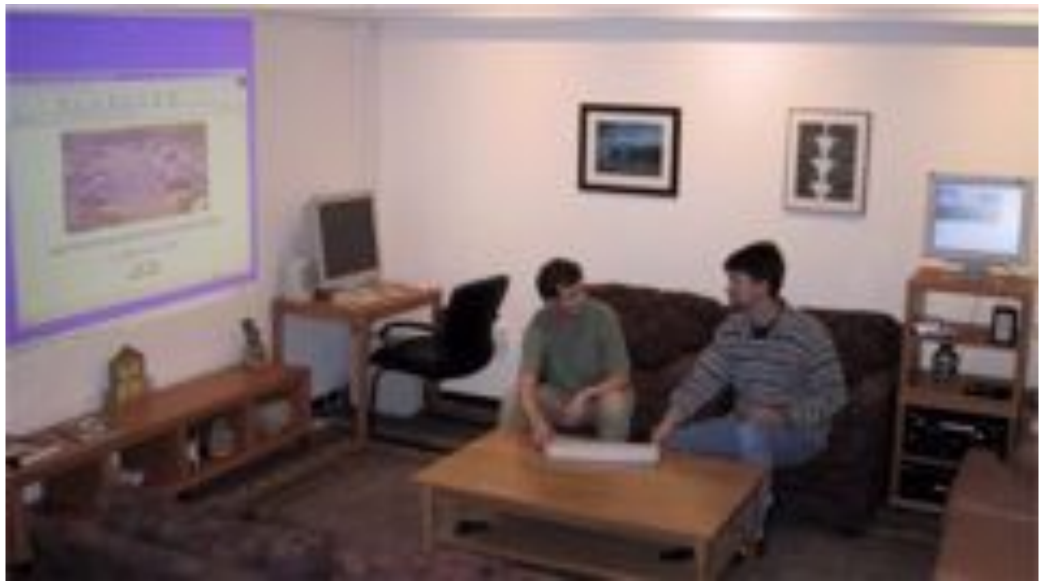
www.embassi.de, 1999–2003

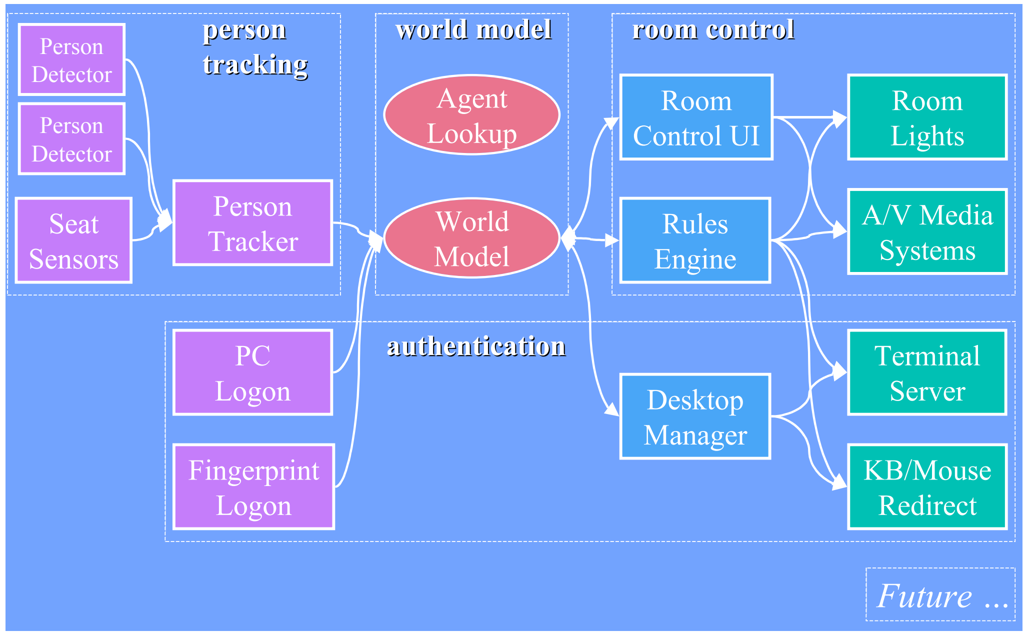
Microsoft's "EasyLiving"

- A prototype architecture and technologies for building intelligent environments.

  Key features include:

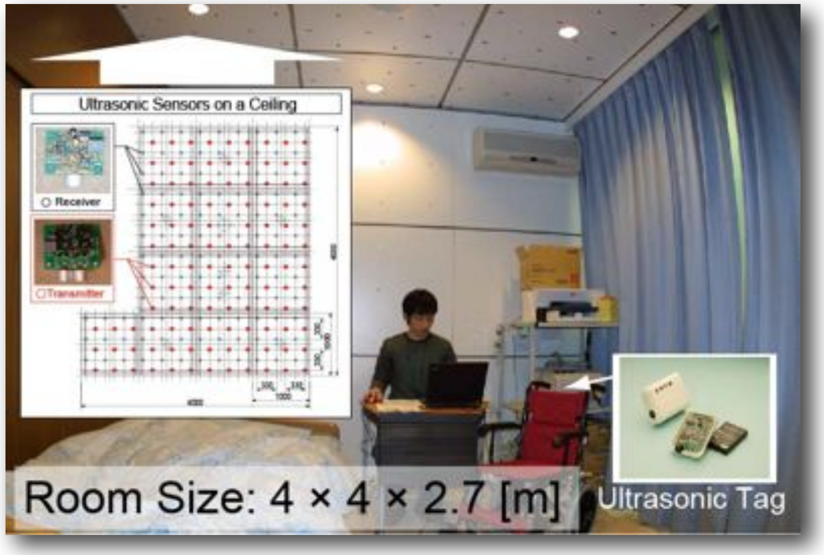
  - XML-based distributed agent system
  - Computer vision for person-tracking and visual user interaction.
  - Multiple sensor modalities combined.
  - Use of a geometric model of the world to provide context.
  - Automatic or semi-automatic sensor calibration and model building.
  - Rule-based definition of automatic behavior.
  - Fine-grained events and adaptation of the user interface.
- Scenarios: Teleporting, Light Control
  - www.research.microsoft.com/easyliving
    [Brumitt et al, 2000]





Numerous other Initiatives

- InHaus (D)
- Aware Home (USA)
- Adaptive House (USA)
- iWork (USA)
- Elder Care Nursing Environment (JP)
- iDorm, MavHome, …



Controlling Dynamic Device Ensembles

Smart Environment Control: Approaches

Project	Intention Analysis	Device Coordination	Coordinator
MavHome, UTA	Learning and Prediction, ALZ	Learned Procedures	Learned from User
The Adaptive House, Boulder	Learning and Prediction, NN	Learned Procedures	Learned from User
The Aware Home, GaTech	Context Widgets; MySQL	Rule Set (manually eng.)	System Designer
Easy Living, MicroSoft	Geometry Model	Rule Set (manually eng.)	System Designer
AIRE, MIT Oxygen	Rule-based Programming	Rule Set (manually eng.)	System Designer
Intelligent Classroom, NWU	Plan Recognition	Rule Set (manually eng.)	System Designer

- Current approaches rely on human assistance for device coordination
- Problems for Ubiquitous Assistance:
  - Can't learn from user – devices are invisible and unknown to user
  - Can't learn from developer – ensemble composition is dynamic and can't be anticipated
- Consequence:
  - Device Ensemble must be able to autonomously generate control strategies
- Challenge: Is this really possible?
  - How to find out what the user wants to be done
  - How to develop a strategy for achieving the required assistance

Background: EMBASSI (bmbf)

- EMBASSI = Electronic Multimedia Operating- and Service-Assistance
  - 19 Partners from industry and academia
  - Oct 1999 – Sep 2003
  - 4.5 M€ funding p.a. (50%), 36 M€ total budget
- Goals:
  - Make networked technical infrastructures of the everyday life usable for everyone
    - Consumer Electronics, Home Environment
    - Automotive
    - Terminals, PoS/PoI
  - Investigate potential architectural and conceptual frameworks
  - In addition:
    - Multimodal / Multimedia components (gesture rec, face rec, avatars, …)
    - Psychological / Usability Evaluations
    - Universal Accessibility
    - Design Tools

EMBASSI

How to represent user needs?

- ~~How to find out what the user wants to be done~~
- How to develop a strategy for achieving the required assistance

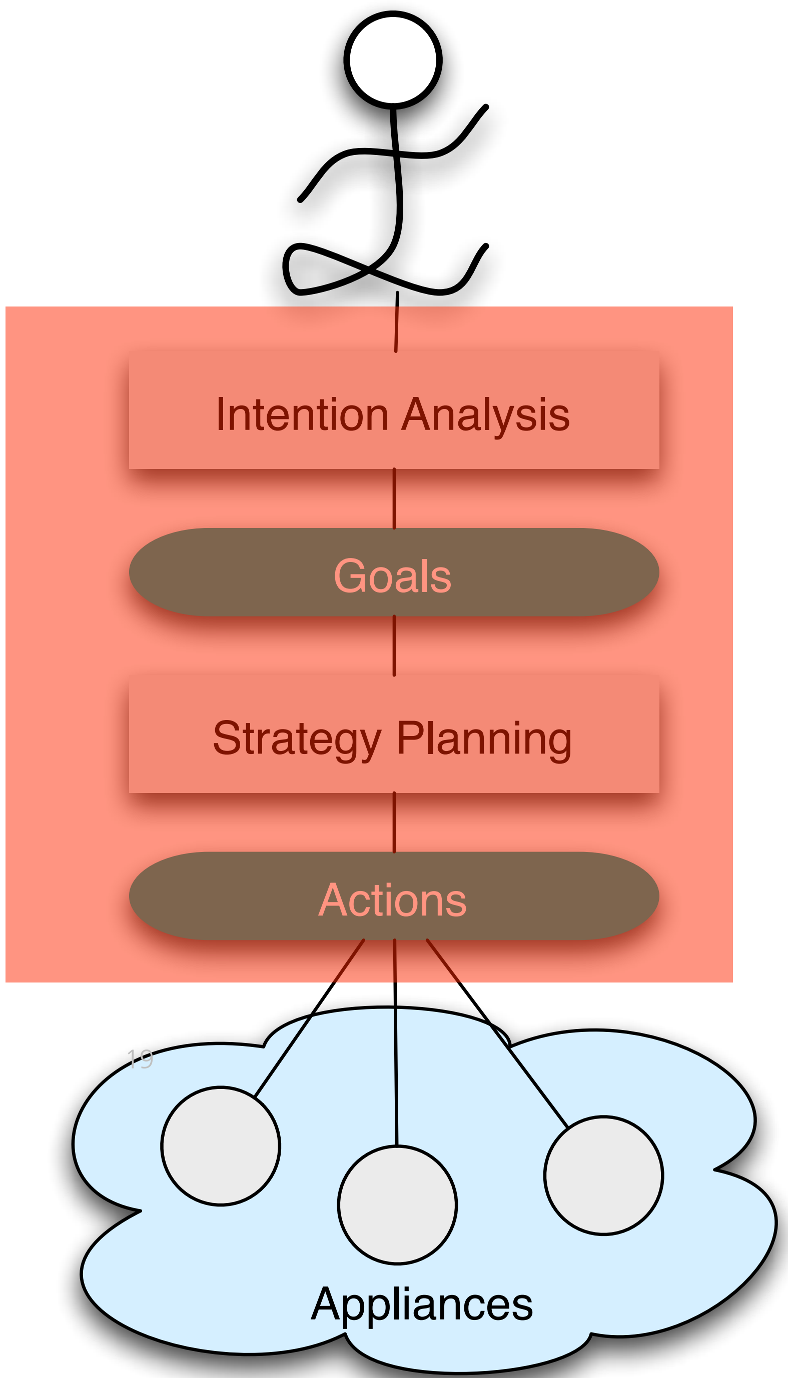
- In general, ensemble membership is unknown to both user (invisibility) and system designer (dynamics)
  - So, user‘s needs can‘t be defined procedurally
    - no way of listing the functions of the devices to execute, if you don‘t know what devices will be available
  - Need a declarative representation of user needs
    - state desired *effects*, *what* to achieve, not *how* to achieve ("Brighter!“)
- Goals:

  Statements about variables of user‘s personal environment that should become true
  - Based on declarative goals, an ensemble may (within limits) autonomously compute appropriate procedures

“Goal-based interaction” [EMBASSI, ~2001]

- 3. Intention Analysis:
  identifies desired goal states (in terms of the state of user‘s personal environment)
  - move from device- to user-oriented vocabulary
- Strategist:
  fills in operations of the devices required for reaching the state
- Two-stage strategist:
  - 1. Select actions
  - 2. Allocate ensemble resources (focus)

How to spread this across an ensemble?



1. Select and Chain Actions

"Brighter" example operators (excerpt)

```
;world model
(:axiom
   :if (not (exists ?x (env-light-contrib ?x)))
   :then (and (not (env-light high))
              (env-light low)))
(:axiom
   :if (exists ?x (env-light-contrib ?x))
   :then (and (not (env-light low))
              (env-light high)))

;lamp's action
(:action turn-down :parameters (?l - lamp)
    :precondition (state ?l on)
    :effect (and (not (state ?l on))
                 (state ?l off)
                 (not (env-light-contrib ?l))))

;shutter's action
(:action close :parameters (?s - shutter)
    :precondition (open ?s)
    :effect (and (not (open ?s)) (closed ?s)))
(:axiom :vars (?s - shutter)
    :if (or (time night) (closed ?s))
    :then (not (env-light-contrib ?s)))


;tv set action
(:action turn-brighter :parameters (?t - tv-set)
    :precondition (or (< (brightness ?t)
                         max-brightness)
                      (env-light low))
    :effect (= (brightness ?t) max-brightness))
```

Approach:

- Provide a common model for the environment state ("Environment Model")
  - This is not trivial … not with 19 partners …
- Describe the semantics of an appliance action as preconditions and effects with respect to the environment model
- Represent goals as constraints on the allowed values of the environment's state variables
- Allows to employ planning algorithms (e.g., partial order planning) for creating multi-appliance strategies for the current ensemble
  - [Heider & Kirste, ECAI'02]

2. Allocate Resources

- Situation:
  - There is more than one device able to perform a given action (such as displaying a multi-medium)
- Challenge:
  - Assigning the task to a device (or *combination* of devices) that will provide the *best* performance in executing the action
  - Resolve *competition* between devices

Device Scenario:
Multi-Display-Environments

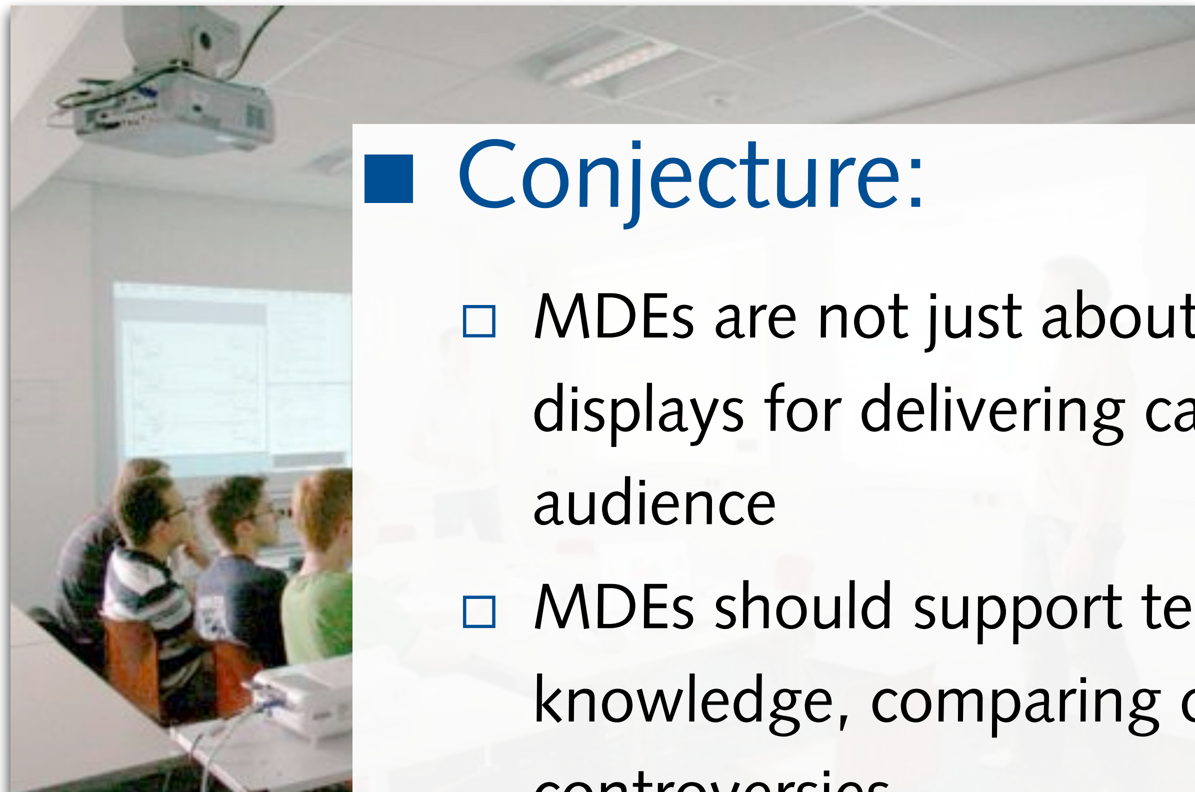
2. Allocate Resources. Scenario: Multi-Display Environments

- Environments with numerous public and private displays
- Active research topic since UbiComp'04
- Challenge:
  - How to allocate documents to displays in order to optimally satisfy users' information needs?
- Question:
  - Is it possible to assist teams in effectively using multi-display environments for working together?
    - without distracting the team from its original task through having to fiddle around with the display infrastructure
  - Main alternative
    - Manual vs. automatic assignment of information to displays

Figure 1. Usability nightmare: multiple screens (five screens of various sizes and shapes, including a podium screen) for using rich media in a conference room. What goes where? How can such a complex environment be designed for maximum usability? Is this design viable across cultures?

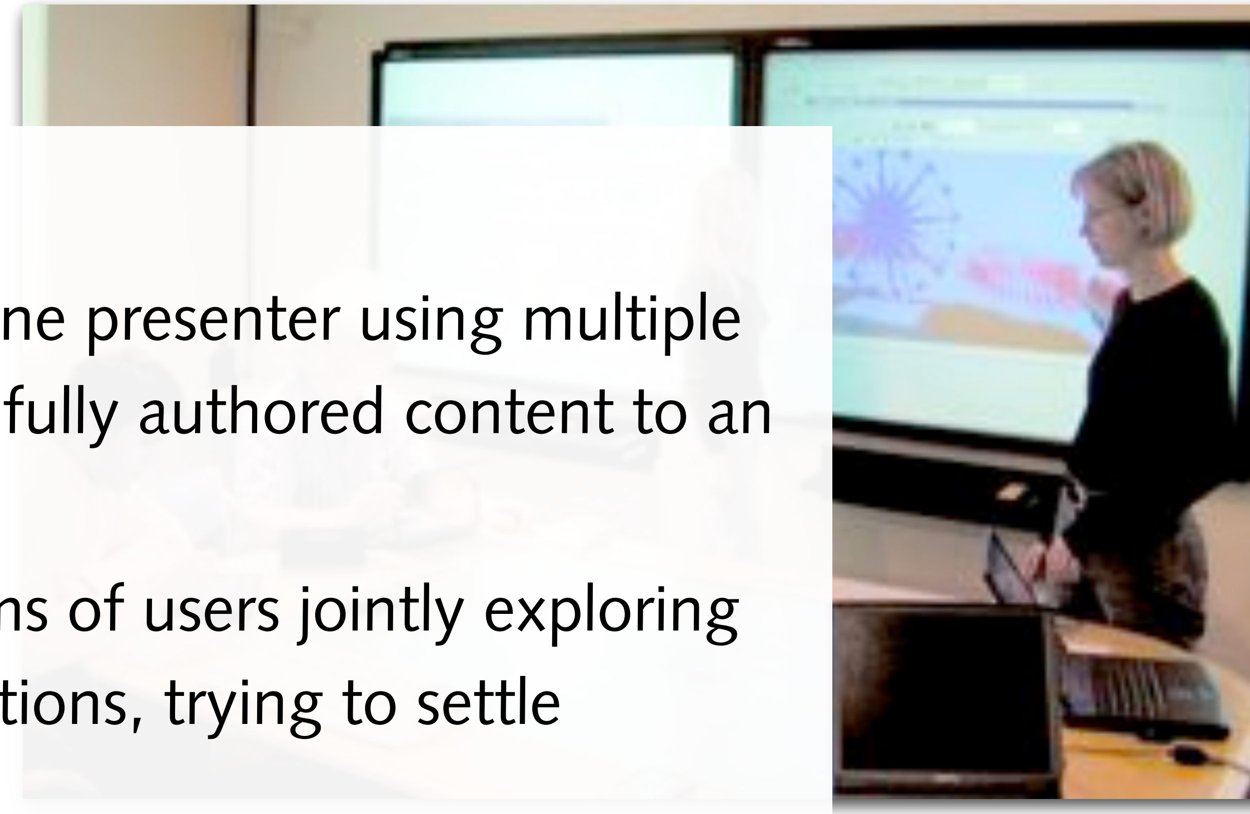
Maribeth Back et al, UbiComp'06 Workshop on Next Generation Conference Rooms

Example Multi-Display Environments

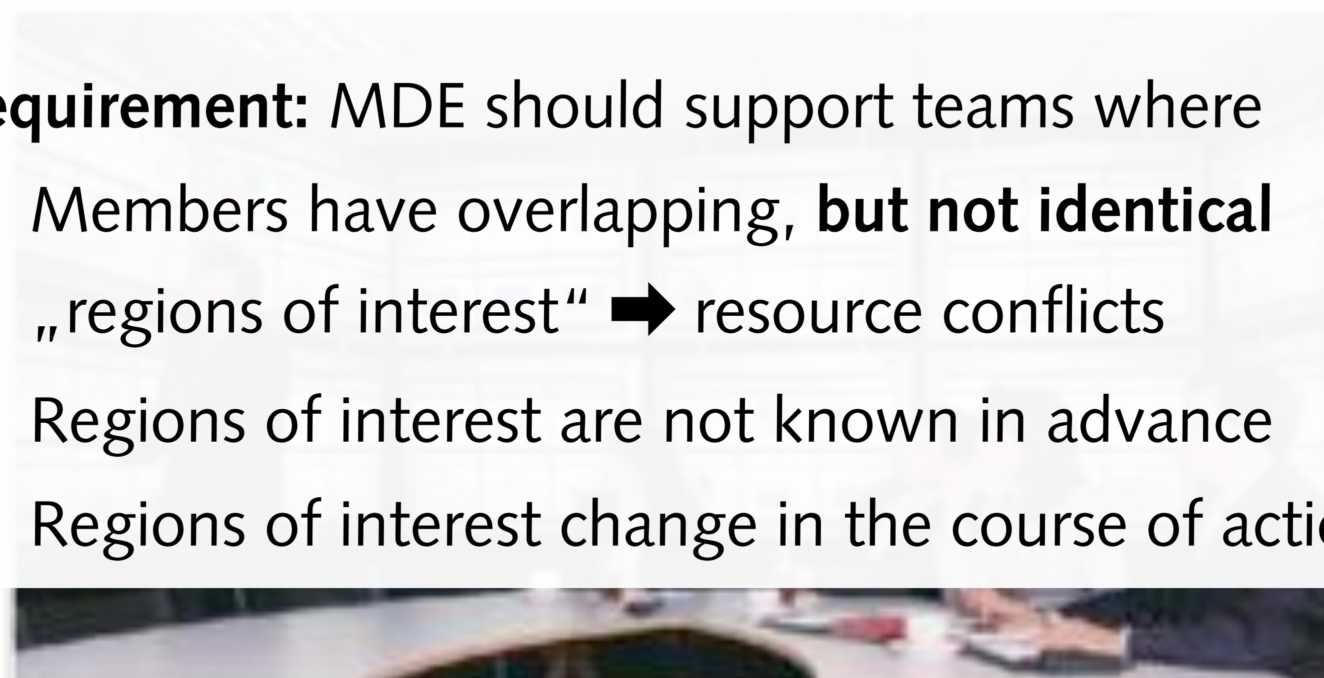


Smart Appliance Lab, Rostock

iRoom, Stanford



- **Conjecture:**
  - MDEs are not just about one presenter using multiple displays for delivering carefully authored content to an audience
  - MDEs should support teams of users jointly exploring knowledge, comparing options, trying to settle controversies

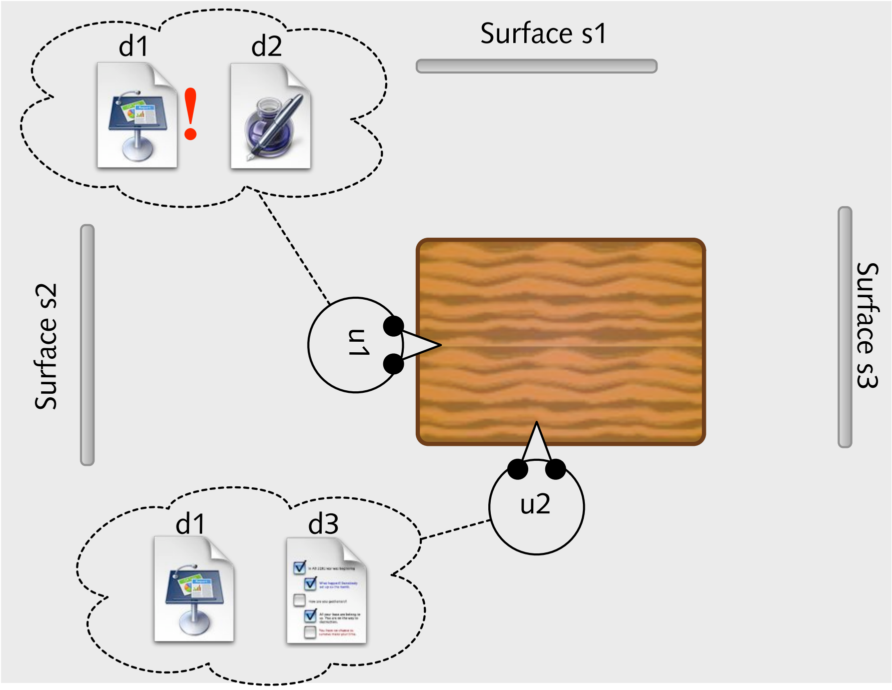


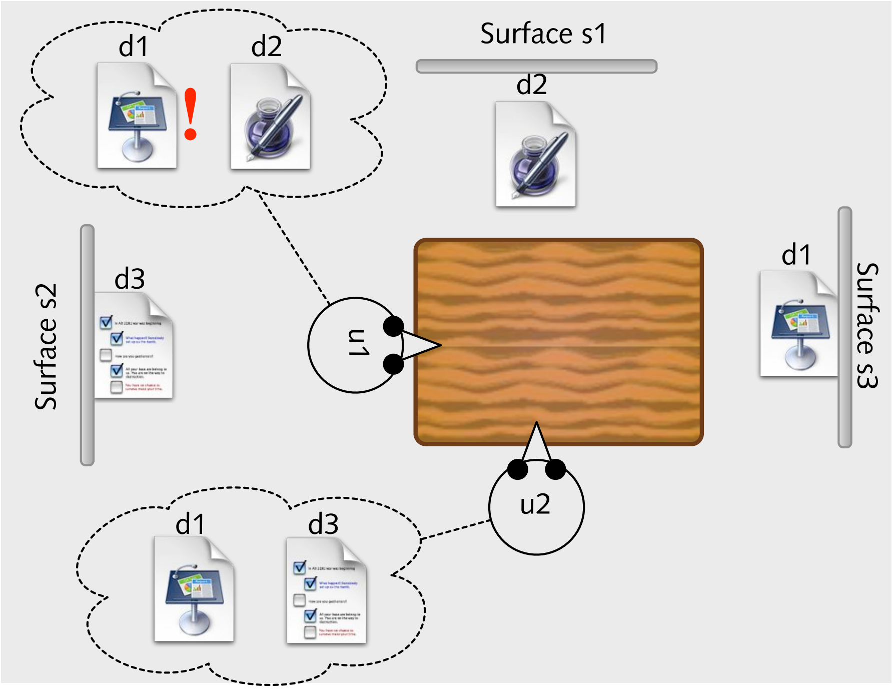
  - **Requirement:** MDE should support teams where
    - Members have overlapping, **but not identical** „regions of interest“ ➡ resource conflicts
    - Regions of interest are not known in advance
    - Regions of interest change in the course of action

“Management Cockpit” at Iglo-Ola, Unilever Belgium

The Display Mapping problem

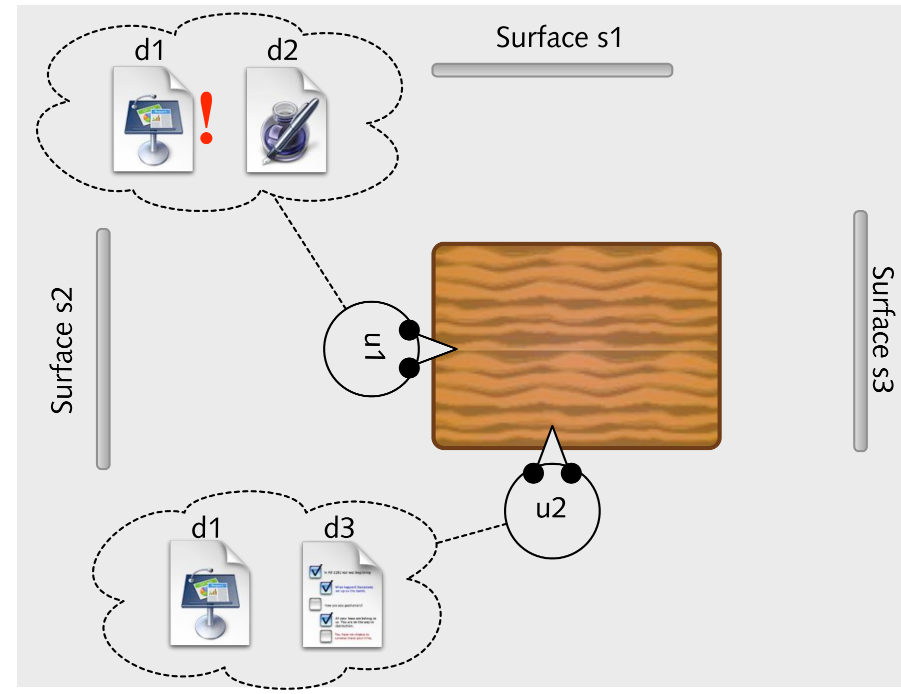
- Example
  - A meeting room with multiple screens (s1-s3), 2 users (u1, u2) and some documents (d1-d3)
  - User u1 is interested in d1 and d2, and u2 is interested in d1 and d3
  - d1 is very important to user u1
- Challenge:
  - find a mapping from displays to documents that maximizes the visibility
- Observation:
  - Assigning resources to actions is an optimization problem

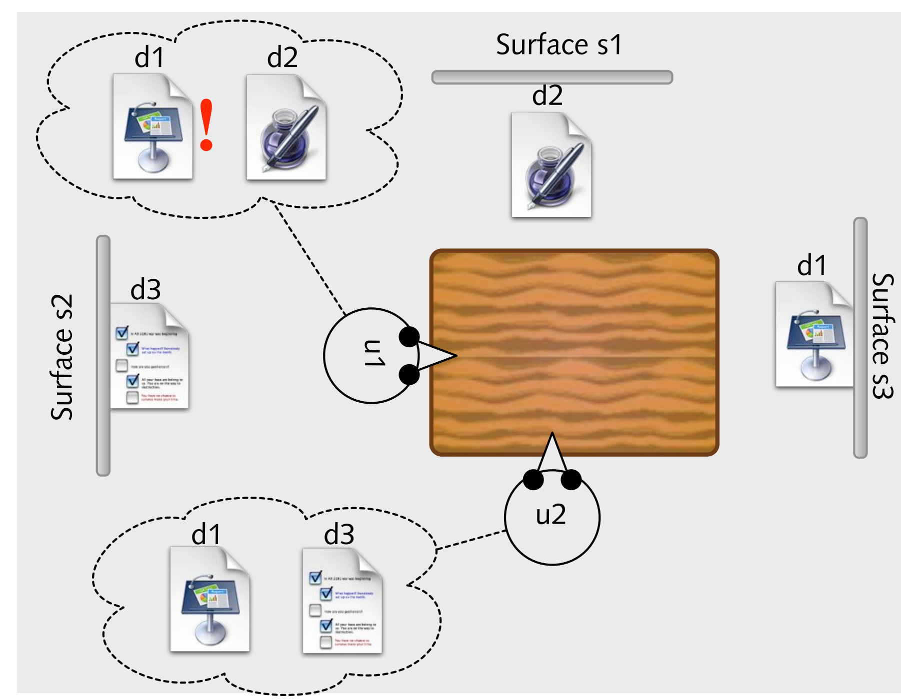




The Display Mapping problem

- Example
  - A meeting room with multiple screens (s1-s3), 2 users (u1, u2) and some documents (d1-d3)
  - User u1 is interested in d1 and d2, and u2 is interested in d1 and d3
  - d1 is very important to user u1
- Challenge:
  - find a mapping from displays to documents that maximizes the visibility
- Question:
  - How to describe the optimization goal?





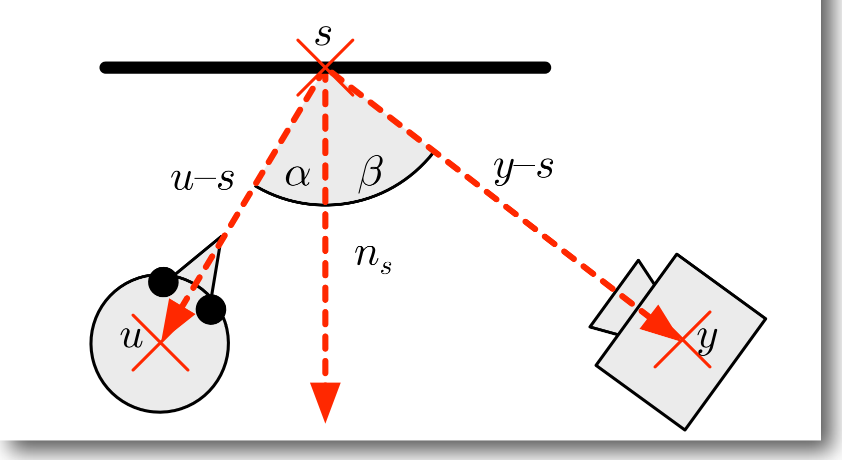
Defining Quality of Display Mappings

- Mapping $m : D \rightarrow 2^{Y},$
  - Maps Documents D to sets of Displays Y
- Spatial Layout, $q_s(m)$:
  - For documents of high importance to a user, displays should be preferred that provide a good visibility for the user (taking into account user & display pos.)
- Temporal Continuity, $q_t(m,m_o)$:
  - When considering a display for a document for a new mapping m, the system should prefer already existing assignments in the current mapping m_o
- Semantic Proximity, $q_p(m)$:
  - Related documents should be presented close to each other to support the user in analyzing the semantic correlation between the documents.
- Overall Quality to maximize:

$$q(m, m_0) = \alpha q_s(m) + \beta q_t(m, m_0) + \gamma q_p(m).$$

Display-Quality: Spatial Layout

$$q_s(m) = \sum_{\substack{u \in U \\ d \in D}} impt(d, u) * \max_{y \in m(d)} vis(y, u)$$



Visibility: vis: Display y × User u → [0…1]

Importance: $impt$: Document d × User u → [0…1]

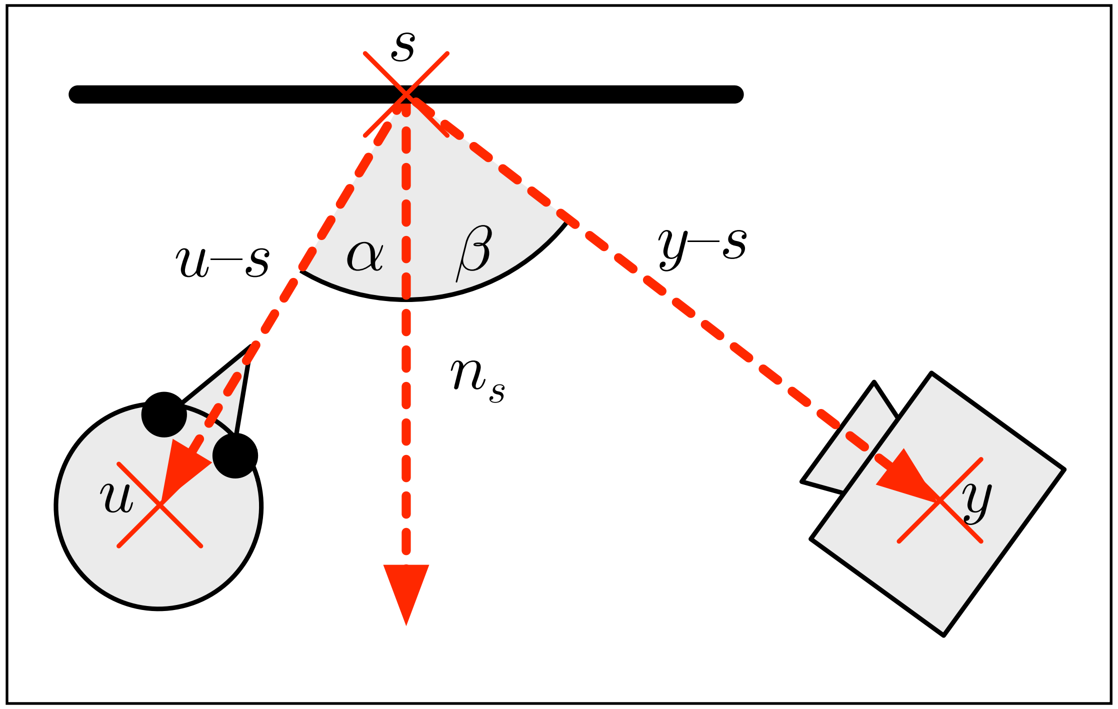
- Documents with a high importance (for specific users) should be assigned to displays with high visibility (for this user)

Spatial Layout, Refined

$$q(dm, ym) = \sum_{\substack{u \in U \\ d \in D}} impt(d, u) * \max_{y \in dm(d)} \big(vis(ym(y), u) * rend(y, ym(y))\big)$$

to optimize, find $(dm_{max}, ym_{max}) = \arg\max_{\substack{dm \in D \rightarrow 2^Y \\ ym \in Y \rightarrow S}} q(dm, ym).$

Why $rend$? Because:



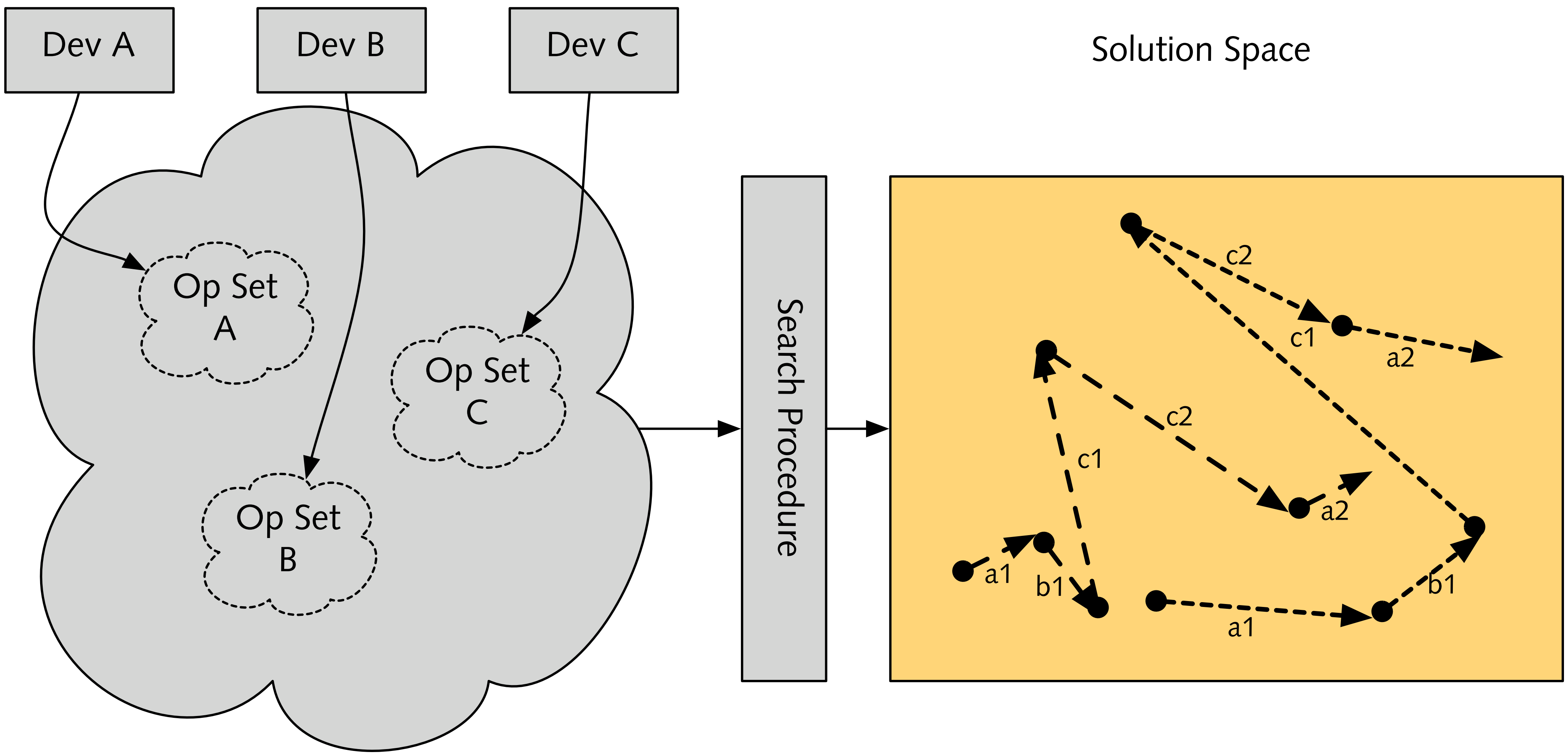
Computing Strategies in an Ensemble

How to distribute strategy generation …

- Naturally distributed problem
  - projectors, screens, notebooks, …
- Can‘t assume central optimizer in a dynamic ensemble
- Can‘t assume global knowledge of device capabilities
  - e.g., each projection surface may have its own specific visibility function (non-Lambert reflectance, …); its own projectability function
  - => can‘t assume a device to understand capabilities & contribution of other devices
- Challenge: How to compute dm_{max}, ym_{max}
  - in a distributed environment?
  - where each device does only need to be able to assess <u>its own</u> contribution to a solution?

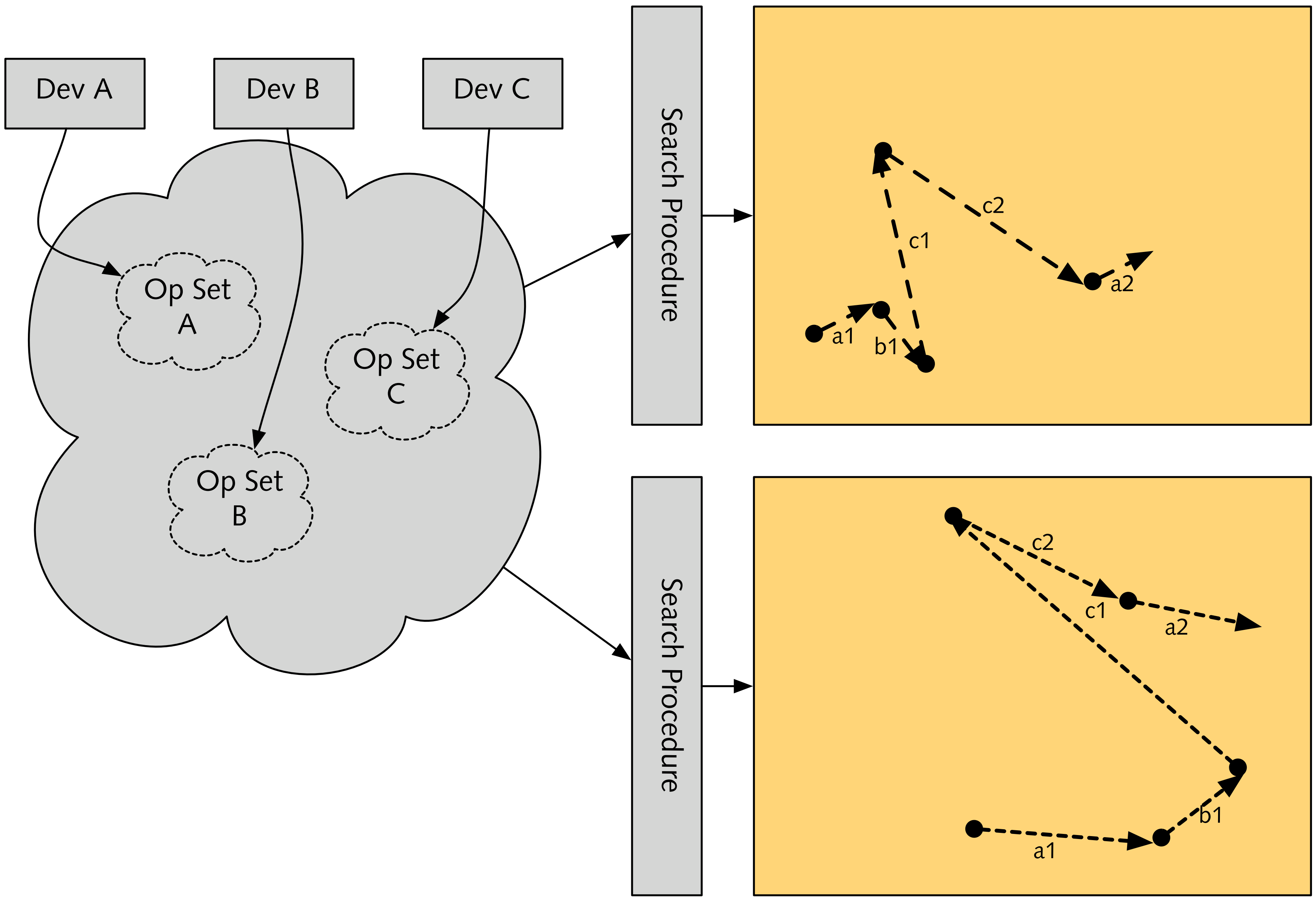
Finding solutions for multiple devices …

- Basic Principle:



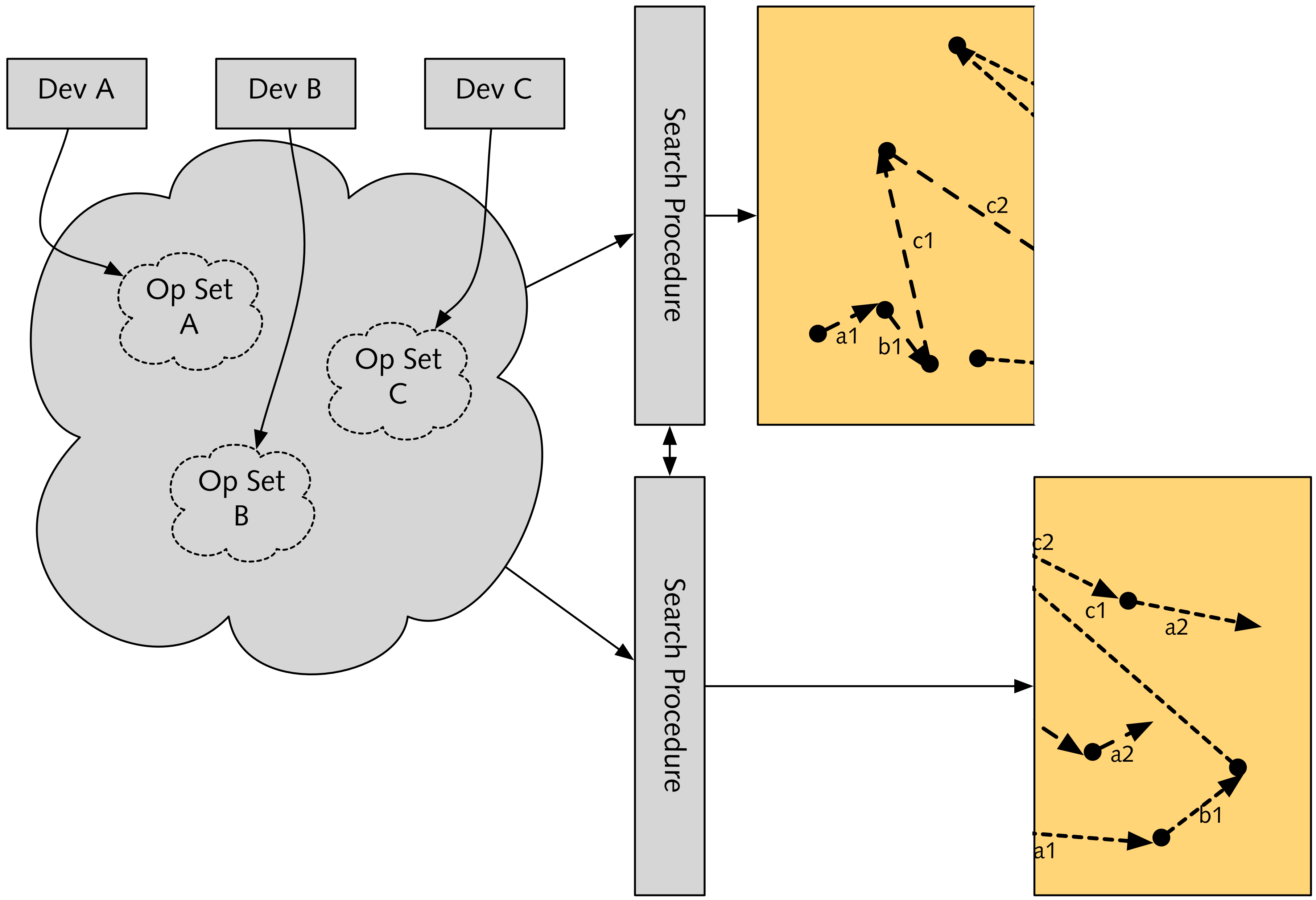
- Try multiple starting points and climb hills

Multiple machines: search in parallel



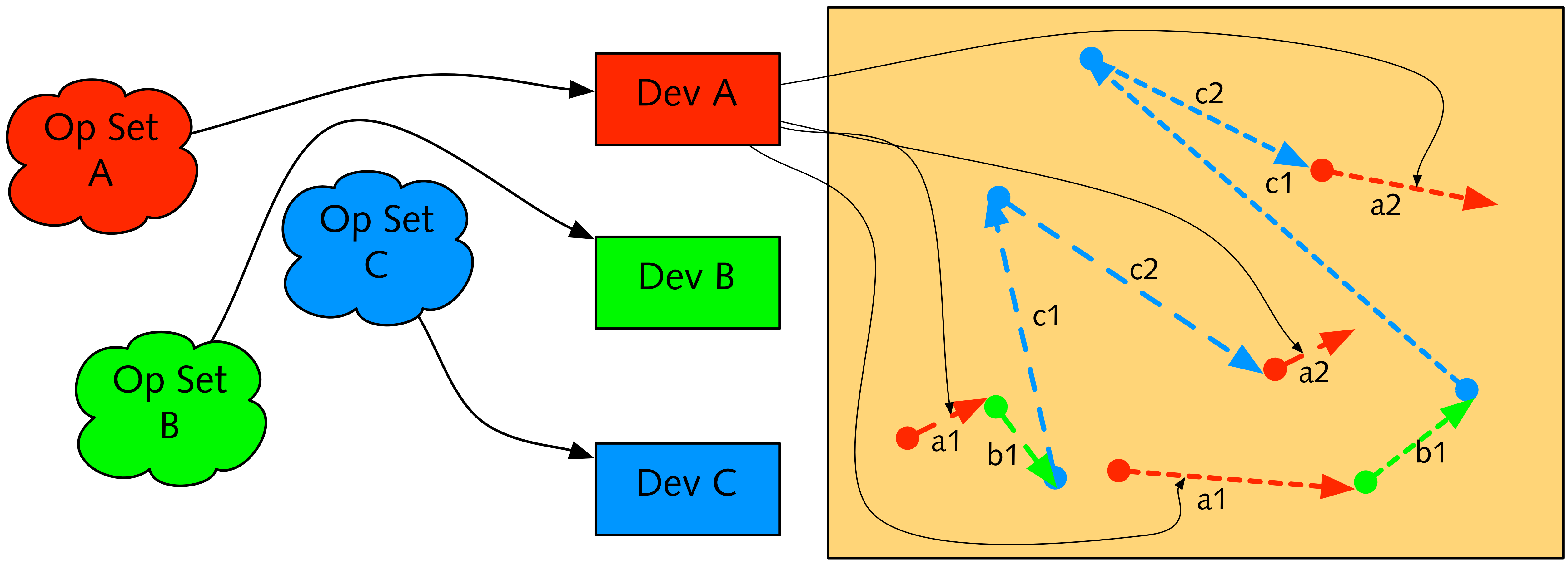
- Still requires to understand global op set in each search procedure instance

… or split search space.



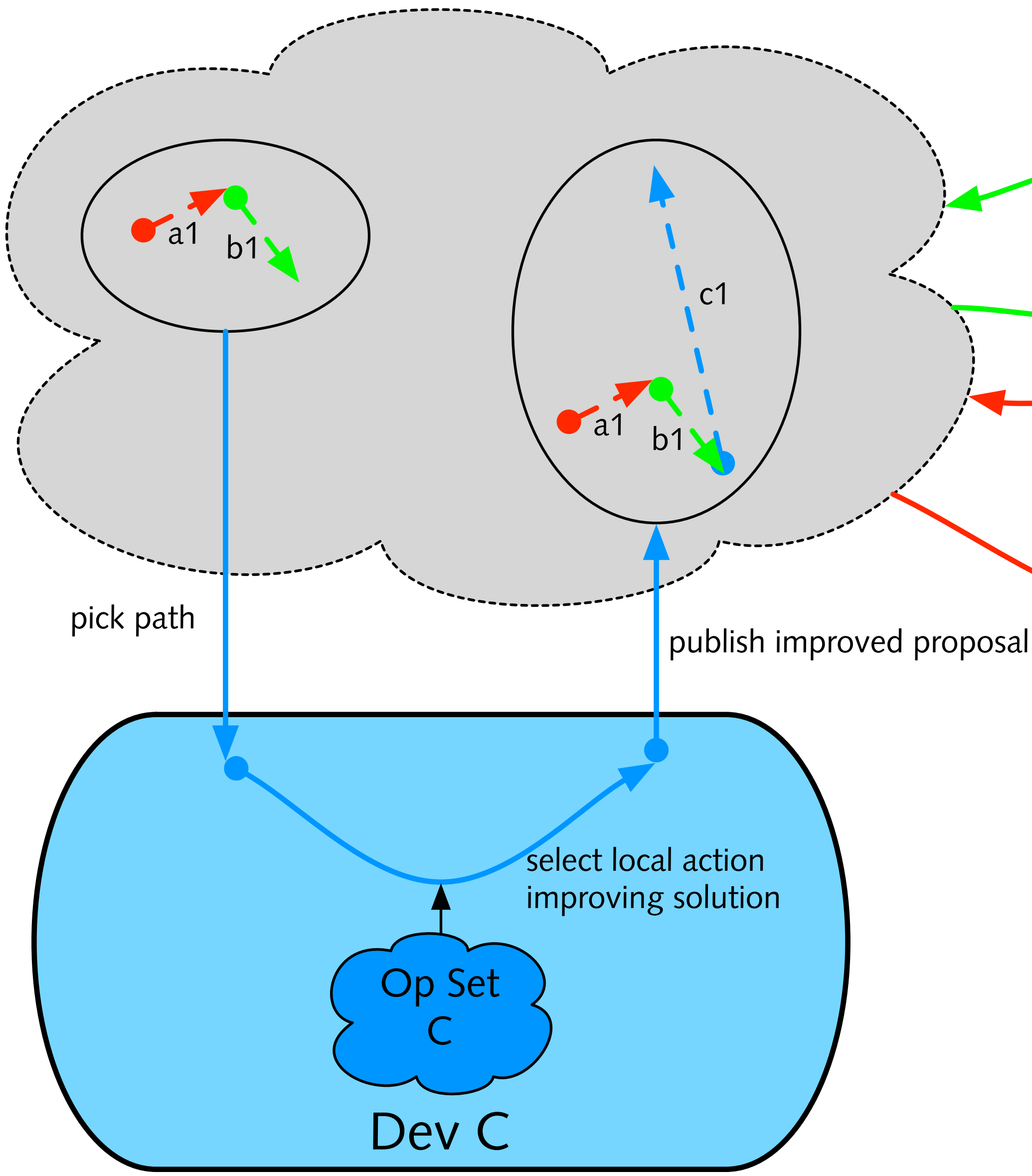
- Still requires to understand global op set in each search procedure instance

What we really would like to have:



- Distribution of search path generation across devices
- Each device only needs to understand its own operations‘ contribution to a solution

Implementation Approach



- DGRASP – Distributed GRASP
  [Heider & Kirste, PDCS'06]
  - Specifically for optimization (Display Mapping)
  - Achievable quality: ~ 98– 99% of max(q)
- Generic concept also usable for action planning
  - Distributed POP
- Major challenge: Exponential growth of paths
  - Which proposal to pick?
    - Best? Shortest? …
  - Which action to select?
    - Greedy? Stochastic?
  - How to publish?
    - To all? (Good in WLAN)
  - When to discard paths?
  - How to learn from previous solutions?

Making Sure That All of This is Any Good

Using Multi-Display Environments

- Environments with numerous public and private displays
- Challenge:
  - How to allocate documents to displays in order to optimally satisfy users' information needs?
- Approach:
  - Provide a formal concept of „optimal display mapping" for multi-display environments
- Prove:
  - Automatic assignment using above concept is better (at least as good as) manual assignment
  - => Do a usability evaluation, comparing manual against automatic

Figure 1. Usability nightmare: multiple screens (five screens of various sizes and shapes, including a podium screen) for using rich media in a conference room. What goes where? How can such a complex environment be designed for maximum usability? Is this design viable across cultures?

Maribeth Back et al, UbiComp'06 Workshop on Next Generation Conference Rooms

Using Multi-Display Environments (Caveat)

- Environments with numerous public and private displays
- Challenge:
  - How to allocate documents to displays in order to optimally satisfy users' information needs?
- **Methodological Problem:**
  - We don't have a really good idea, how users would like to interact with such environments
  - There is no empirical base of use cases
  - We don't know which kinds of conflicts w
    in such environments with what frequ
- Solution Alternatives:

  (A) Wait for organizational psycholog
  experts to investigate this setting

multiple screens (five screens of
ding a podium screen) for using rich
What goes where? How can such a
esigned for maximum usability? Is this
tures?

Maribeth Back et al, UbiComp'06 Workshop on Next Generation Conference Rooms

This can go utterly wrong.

But at least it is a starting point for further discussion.

How to Solve the Display Mapping Problem

… what we tried:

- Find a computable definition for „globally good mapping“ – with respect to
  - Multiple users, multiple documents, multiple displays, different regions of interest and dynamic topic changes
- Build a system that is able to compute a good mapping based on this definition
- Think of an experimental scenario in which to evaluate system performance
- Compare automatic and manual mapping in this scenario
  - Start with a weak claim: try to be as least as good as manual mapping

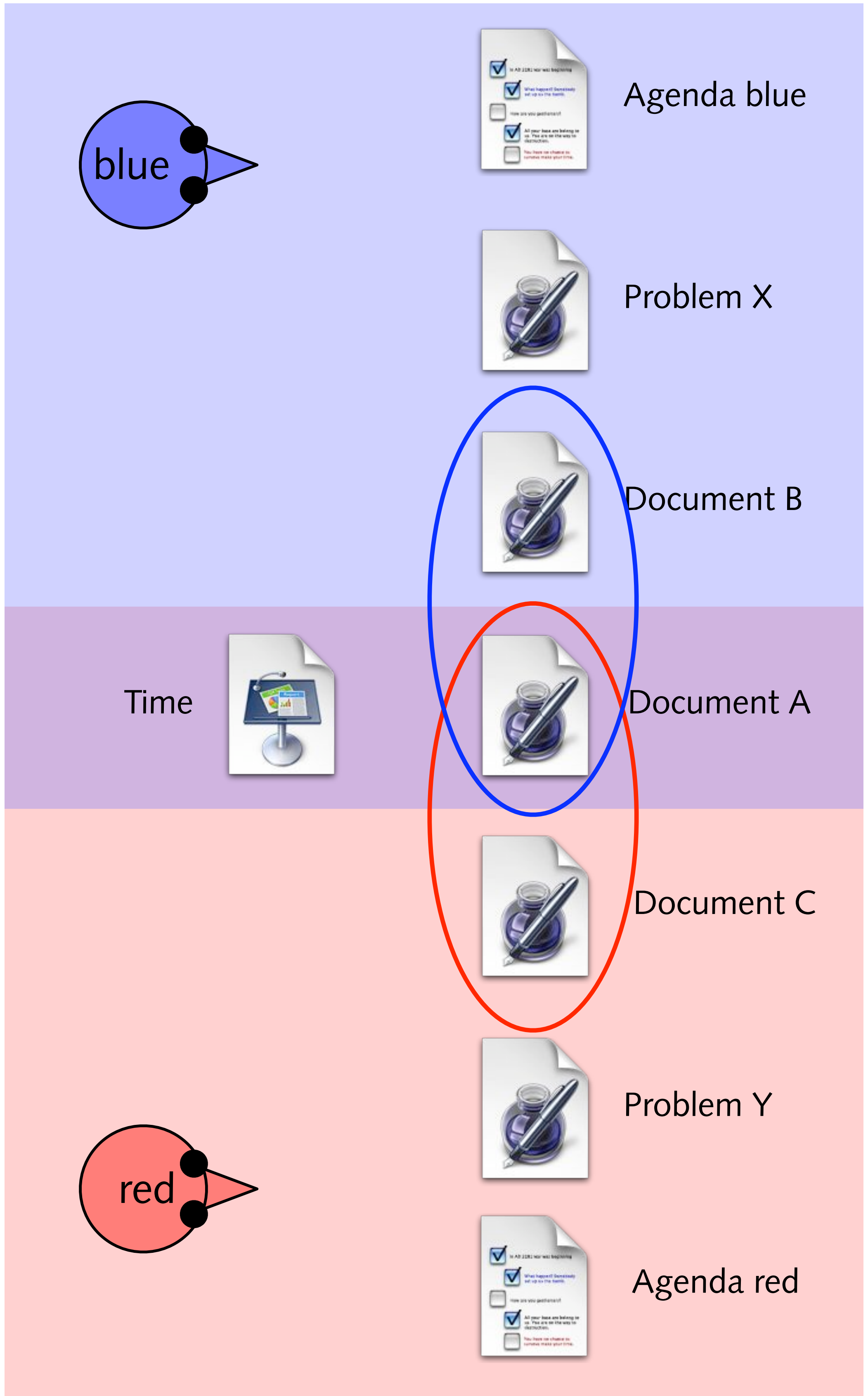
The Goal of the Experimental Design

- Recall: Our conjecture of MDE requirements:
  - Members have overlapping, but not identical „regions of interest“
    ➡ resource conflicts (competition for display space)
  - Regions of interest are not known in advance
  - Regions of interest change in the course of action
- Objectives of experimental design
  - Deliberately induce conflicts between the team members regarding the use of the available display space
  - Enforce substantial changes in the set of documents currently important for a user
  - Allow to compare systems using an objective quantity (performance) rather than a subjective quantity (“user satisfaction”)
  - **Non-objective:** choose an experimental setup that everybody immediately can relate to based on everyday experience …

Experimental Design: Basic Concept

- Have a team solve a joint task
- Quantify performance by measuring the time required for task completion
- Choose an artificial, simple task
  - **simple:** Avoid performance differences being swallowed by "noise" (i.e., by variance in basic task solution time – more complex tasks produce higher absolute noise)
  - **artificial:** Reduce noise contribution from different skills / prior knowledge
- Task design:
  - Semi-cooperative two-person task (assignment: solve joint task as fast as possible)
  - Connection between individual sub-tasks is not known in advance, has to be discovered by the team
  - Individual tasks: Comparison between letter sequences in two documents
    - requires both documents to be visible to user

Experimental Design: Regions of Interest



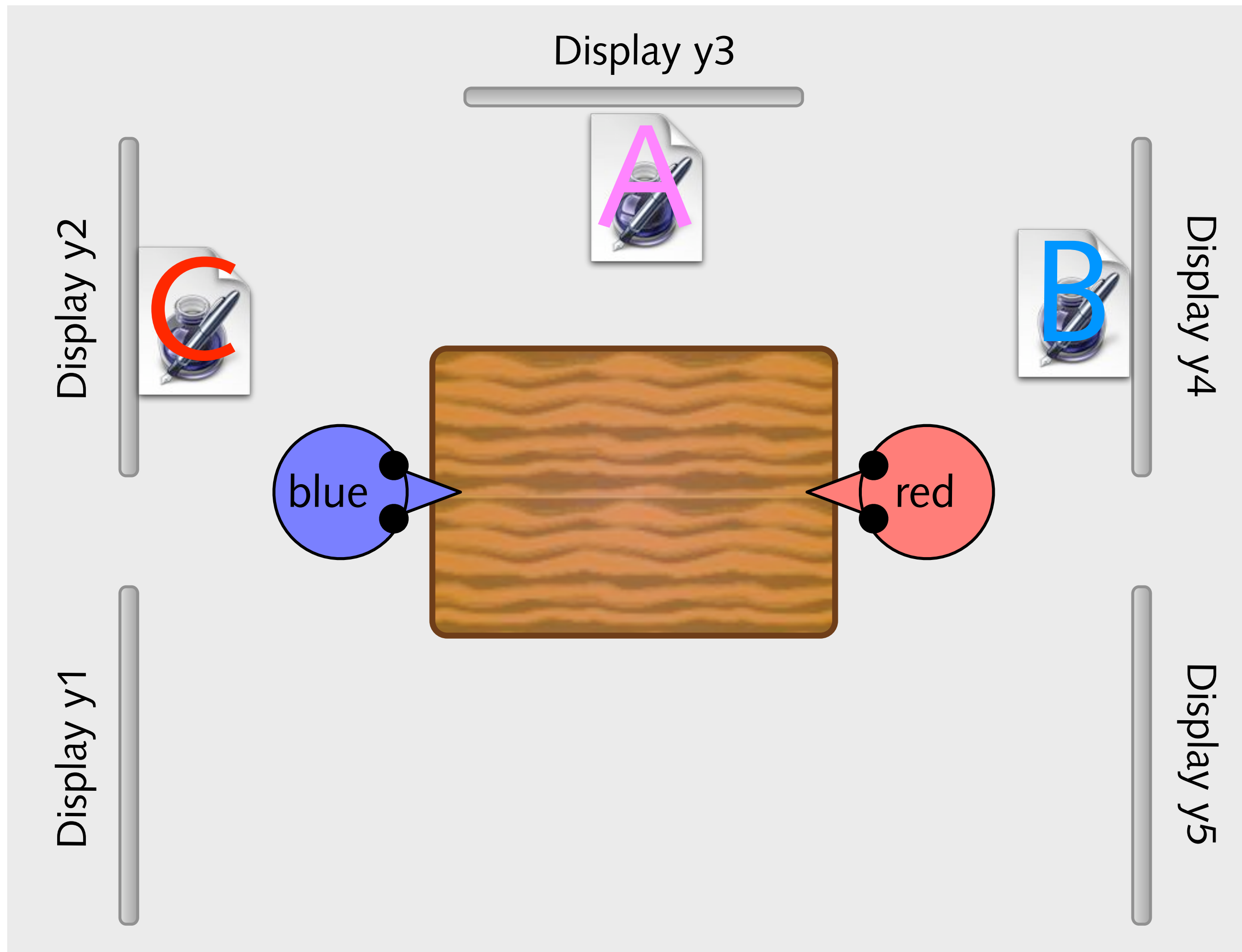
Agenda blue

- Display Time.pps and write down the time and the Key
- Display Problem-X.pps
- Finish the described problem
- Display Time.pps and write down the time

Problem X

- Compare the Letter sequences from file A.pps and B.pps and find all differences
- Record the differences together with the position in the format:
  - 7. G - Q (position, letter in A, letter in B)

Experimental Design: Physical Setup



Solution to be discovered

- 5 Displays (2 individually visible, one shared)
- Enforcement of conflicts: each document can be displayed only on one display
  - In order to simplify experimental setup – otherwise additional shared documents would have to be introduced
- Participants:
  - 24 voluntary subjects (19 male and 5 female)
  - 12 Teams in 2 Groups

Group A:

First Task Set (Initial Test)	Second Task Set (After Training)
Automatic	Manual

Group M:

First Task Set (Initial Test)	Second Task Set (After Training)
Manual	Automatic

Experimental Design: Problem Documents

Agenda blue

- Display Time.pps and write down the time and the Key
- Display Problem-X.pps
- Finish the described problem
- Display Time.pps and write down the time

Problem X

- Compare the Letter sequences from file A.pps and B.pps and find all differences
- Record the differences together with the position in the format:
  - 7. G - Q (position, letter in A, letter in B)

Time

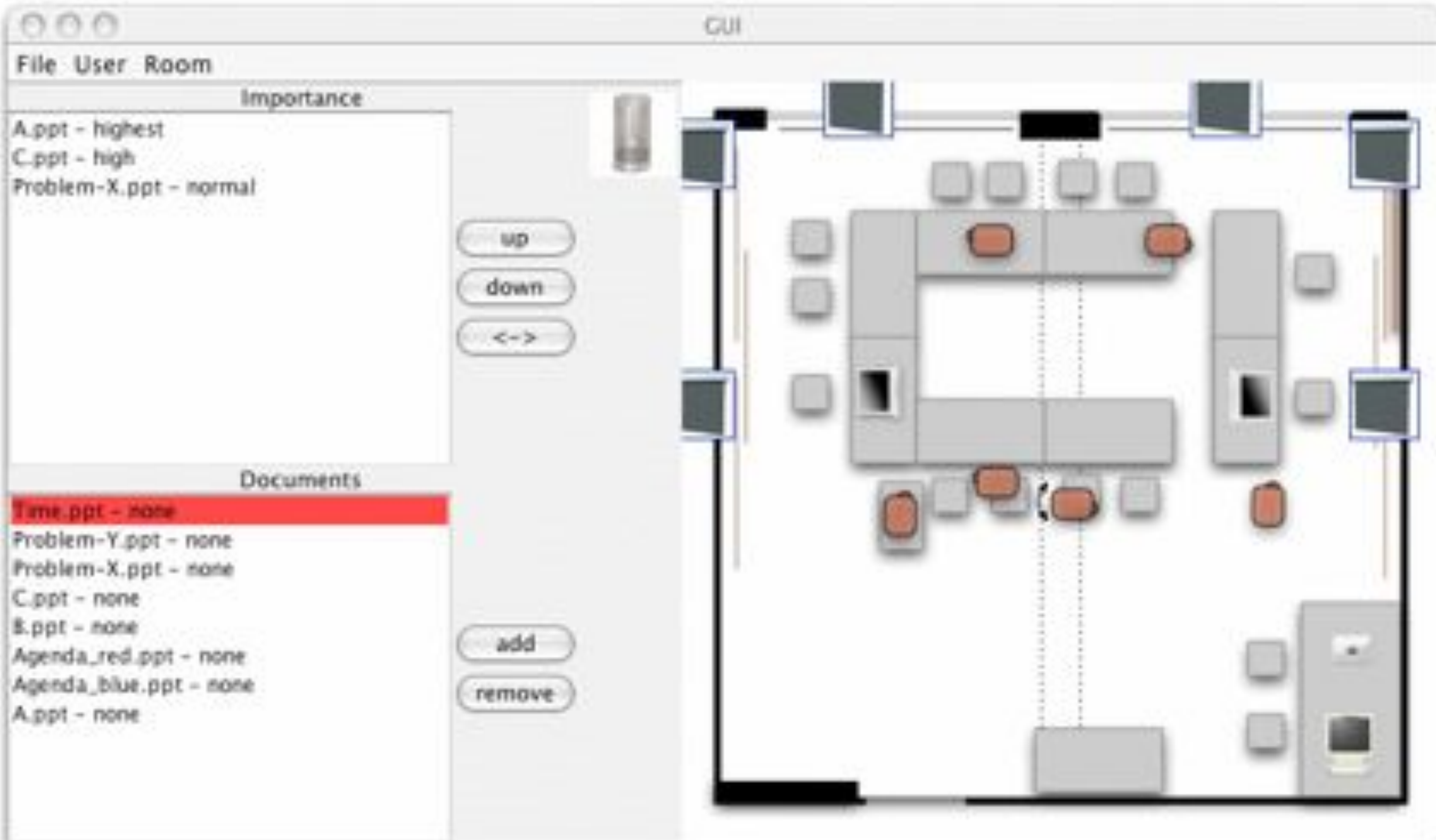
09:02:35

Key:XY83jfgK34Lfb

Letter sequence A

mkFfhuGmeFgvdHnkjEpOokzRkcnhNUgtfdEdfukuNhfcvtG

GUI for Importance Value (Automatic A.) and for Manual Assignment



Experimental Design: Real World Setup

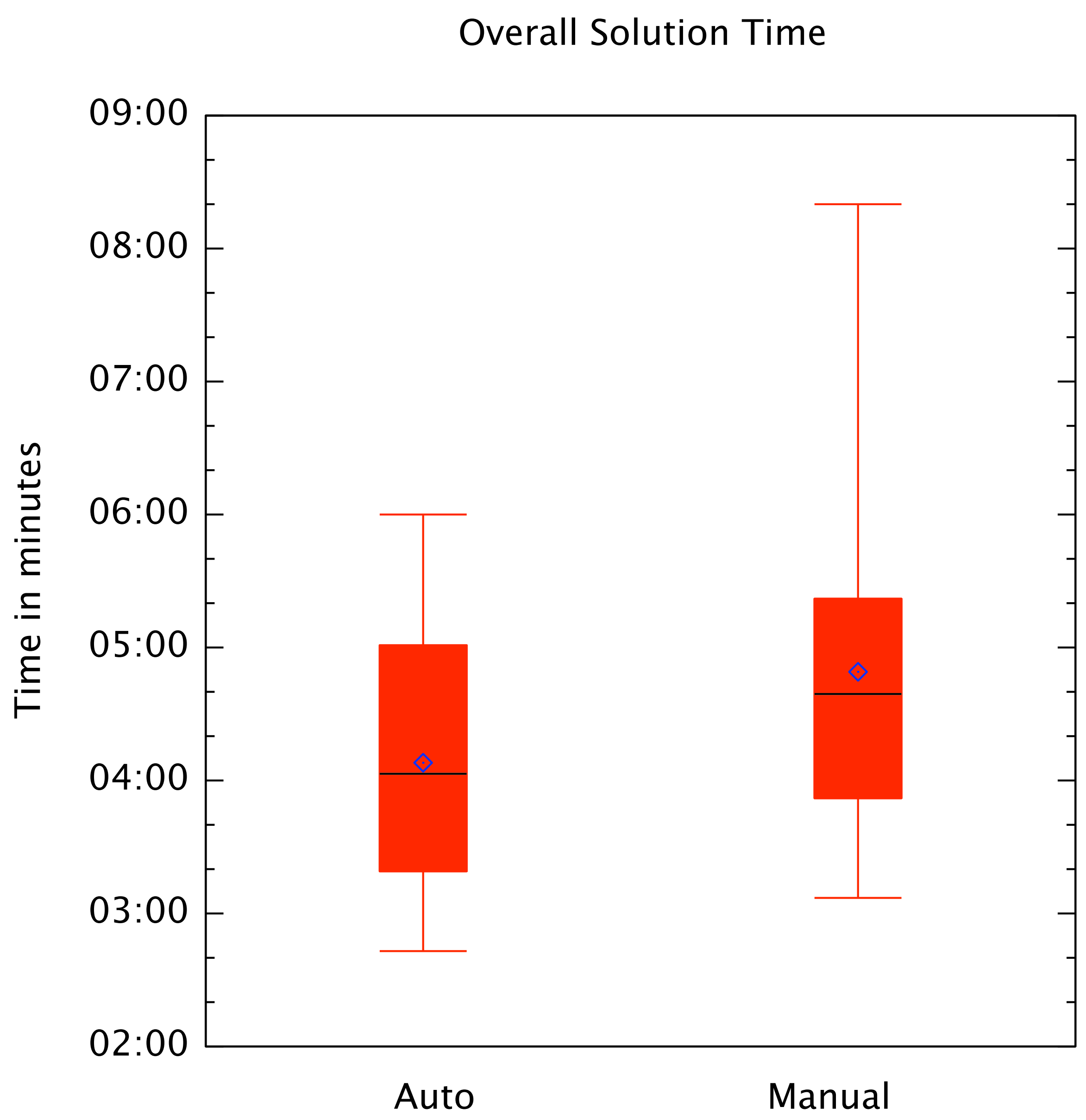
Experimental Design: Real World Setup

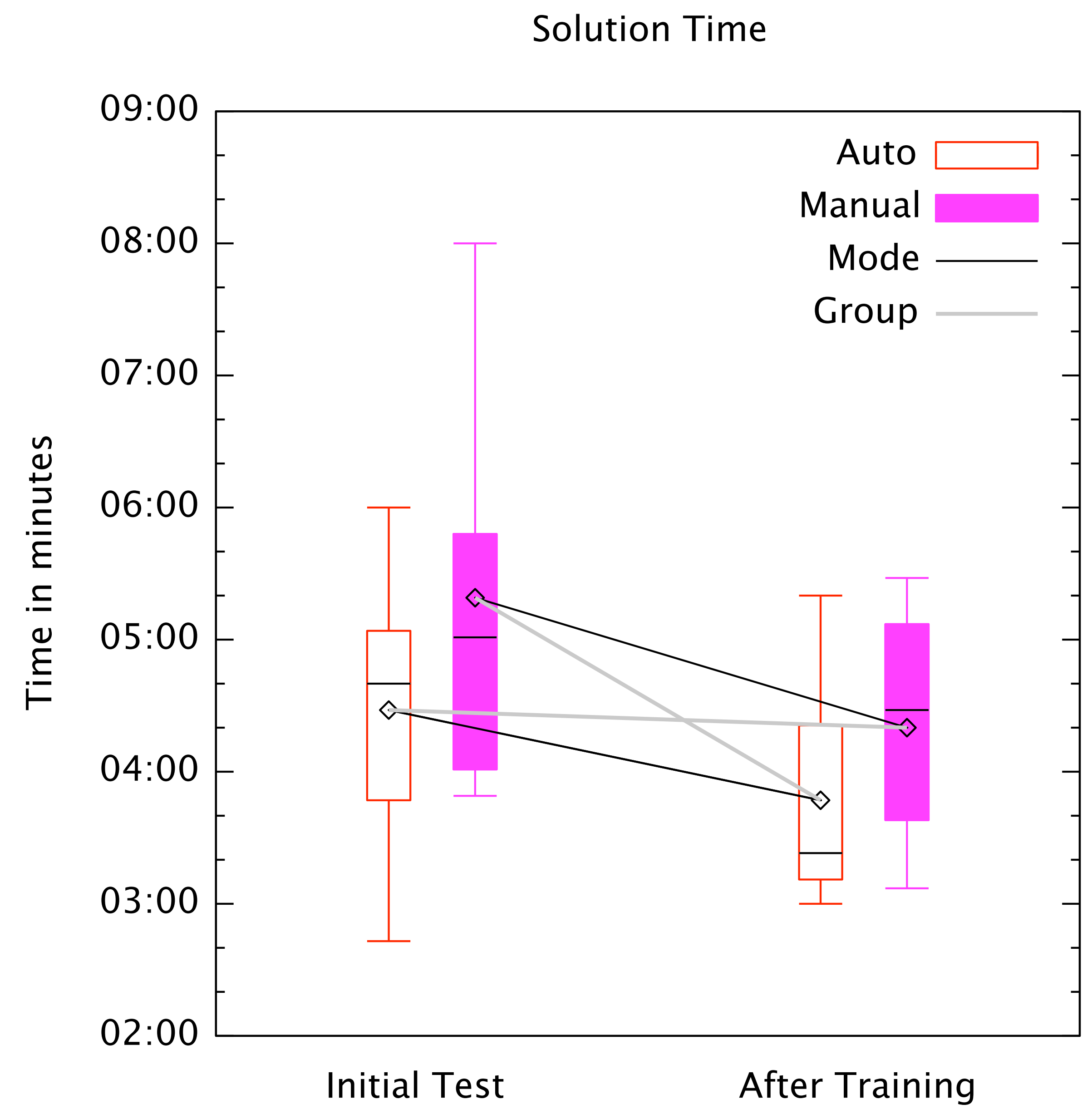
Experimental Design: Recorded Data

- Time required for completing the task
- The number of interactions with the provided user interfaces (number of mappings)

- After each task set, the subjects were asked to answer a questionnaire regarding user satisfaction
- A final questionnaire regarding the comparison of the automatic and the manual assignment

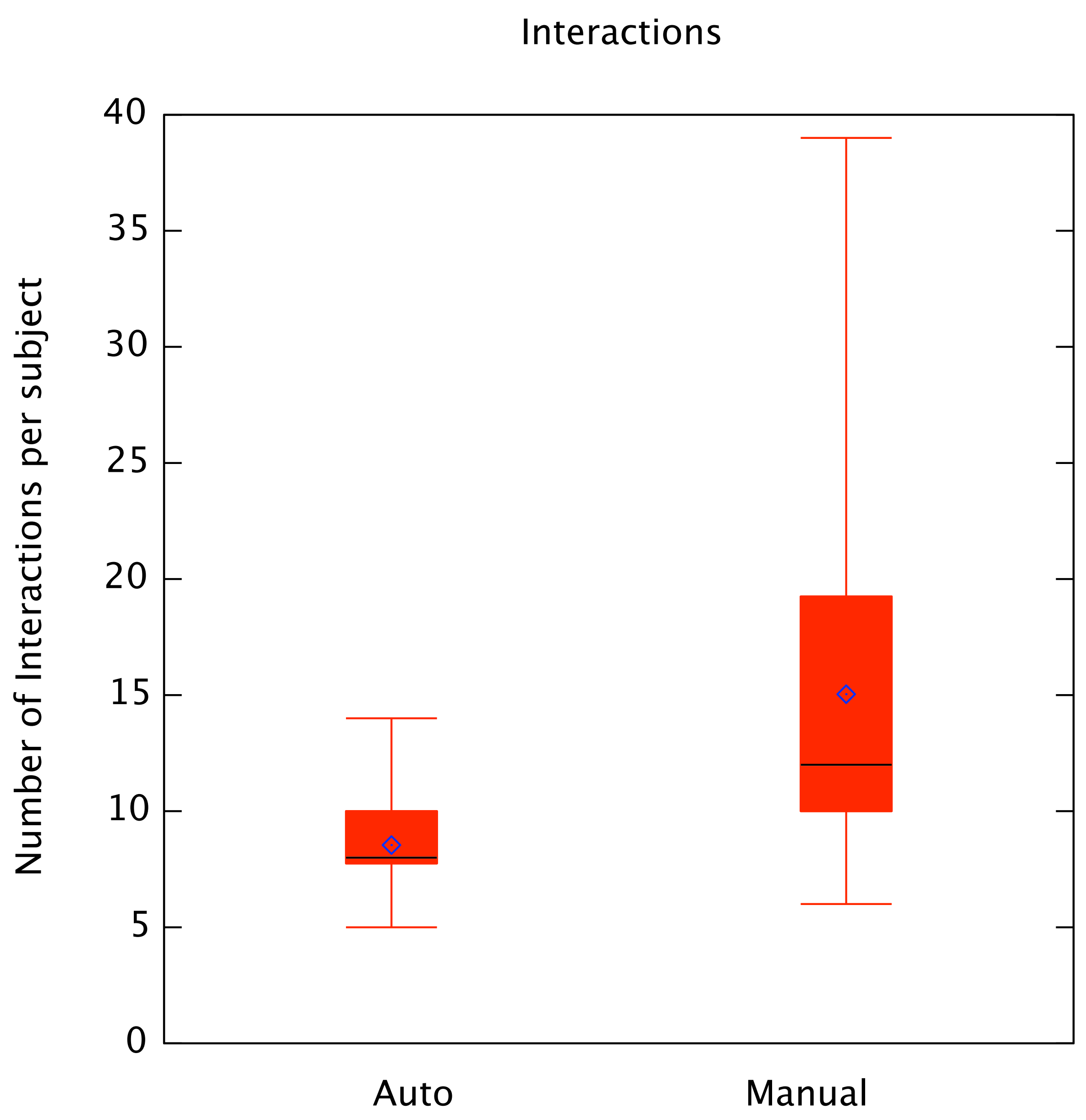
Findings: Solution Time

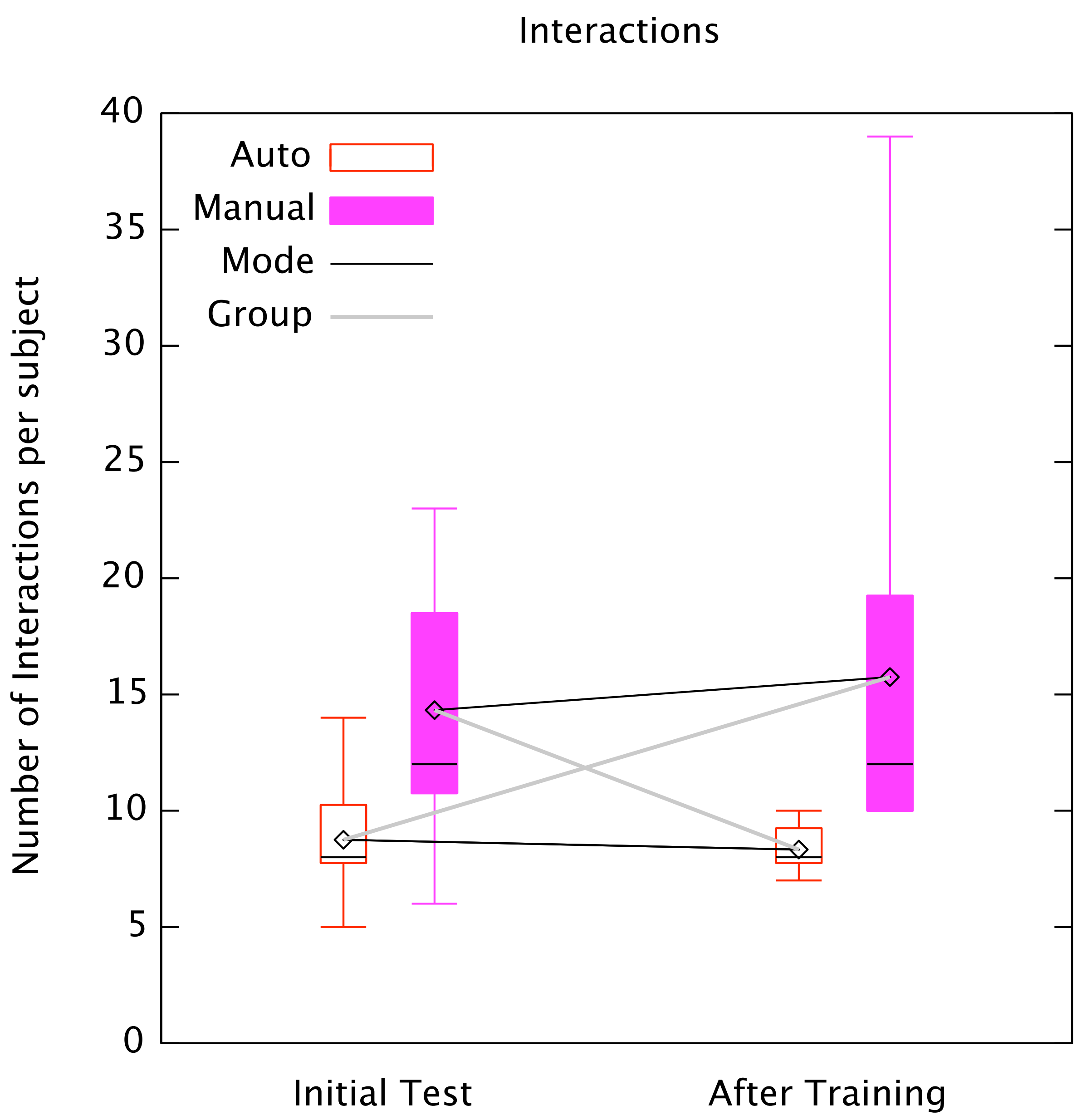




- Automatic assignment: 4:08 min
- Manual assignment: 4:49 min

Findings: Interaction





- Automatic assignment: 8.5 interactions
- Manual assignment: 15 interactions

Findings: Questionnaire ("User Satisfaction")

- Items:
  - The system is easy to use.
  - The system helps in solving the task efficiently.
  - It is easy to cooperate with the team partner.
  - The system helps in solving team conflicts.
  - I felt comfortable in using the system.

A = Automatic, M = Manual, 1 = strongly disagree, 5 = strongly agree.
C = Comparison, 1 = Manual strongly preferred, 5 = Automatic strongly preferred

Item	Group A			Group M			All Participants		
	A	M	C	A	M	C	A	M	C
Ease of use	3.9	4.3	3.3	4.2	3.5	4.1	4.0	3.9	3.7
Efficiency	4.1	3.9	3.2	4.2	3.0	4.3	4.1	3.5	3.7
Cooperation	3.7	3.4	3.8	4.0	2.6	4.1	3.9	3.0	4.0
Conflicts	4.2	1.5	4.5	4.1	1.7	4.4	4.1	1.6	4.4
Comfort	3.6	3.7	2.6	4.1	3.1	3.7	3.9	3.4	3.2
Average	3.9	3.4	3.5	4.1	2.8	4.1	4.0	3.1	3.8

Findings: Summary

- With Automatic Mode
  - Solution time is smaller (= cognitive load / distraction is lower)
  - Number of interactions is smaller (less work for conflict resolution)
  - User satisfaction is higher (specifically: less perceived conflicts)
- Empirical evidence for claim
  - Automatic display assignment is more efficient than manual assignment in multi-user, multi-display situations with conflicting and dynamic document sets
- Significance of results:

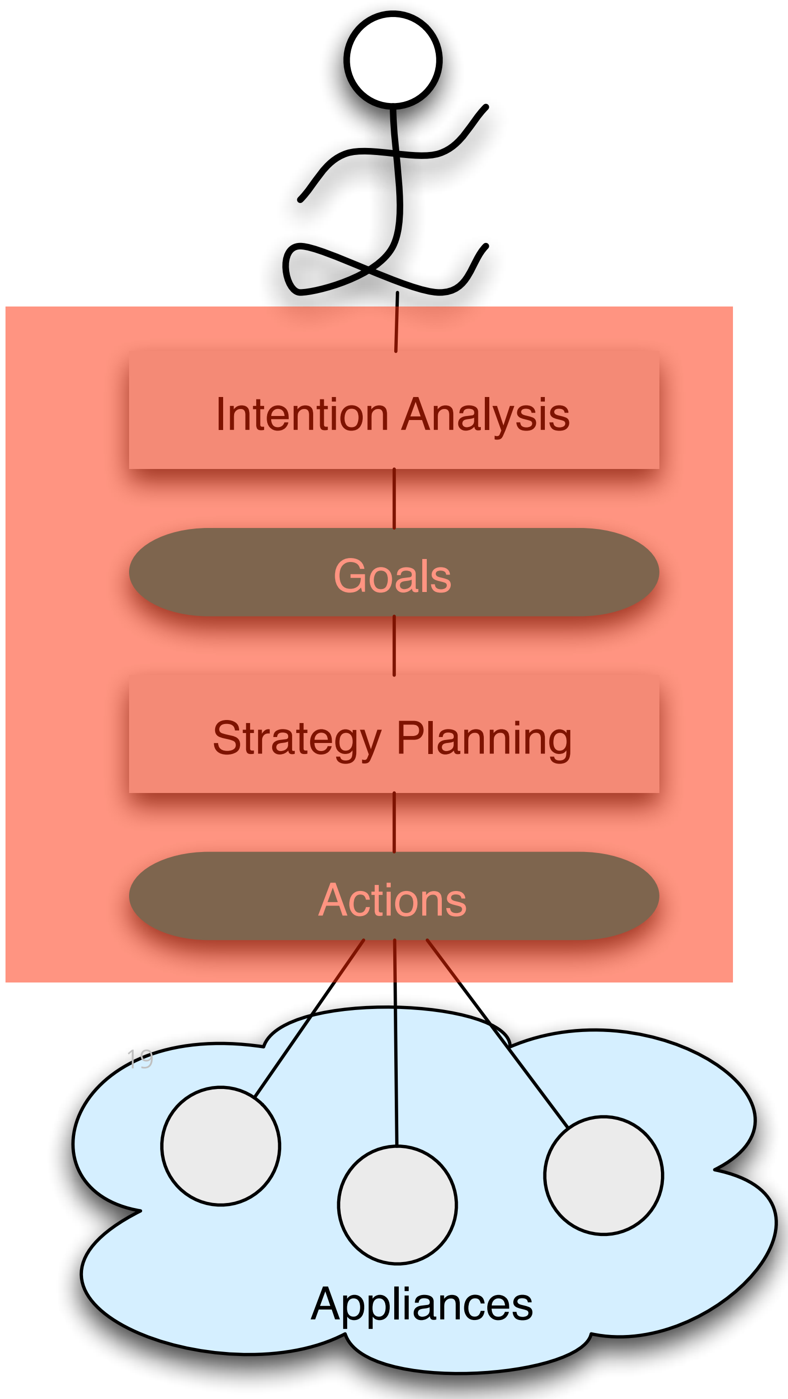
H_0	H_1	H_0 rejected at level
$t_{man} \leq t_{auto}$	$t_{man} > t_{auto}$	2.5%
$i_{man} \leq i_{auto}$	$i_{man} > i_{auto}$	0.5%
$s_{man} \geq s_{auto}$	$s_{man} < s_{auto}$	0.5%

Finally, Intention Analysis

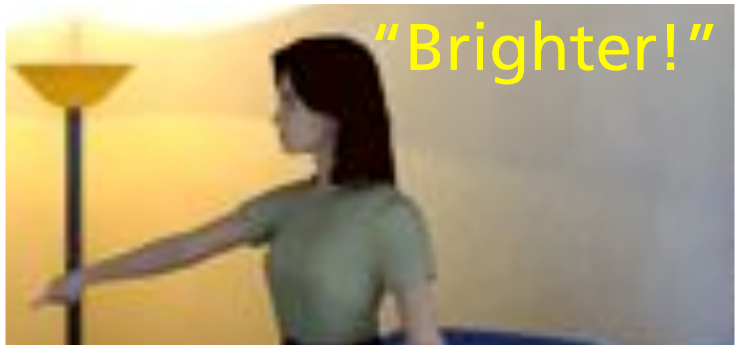
“Goal-based interaction”

- 3. Intention Analysis:
  identifies desired goal states (in terms of the state of user‘s personal environment)
  - move from device- to user-oriented vocabulary
- Strategist:
  fills in operations of the devices required for reaching the state
- Two-stage strategist:
  - 1. Select actions
  - 2. Allocate ensemble resources (focus)

How to spread this across an ensemble?



3. A Note on Intention Recognition

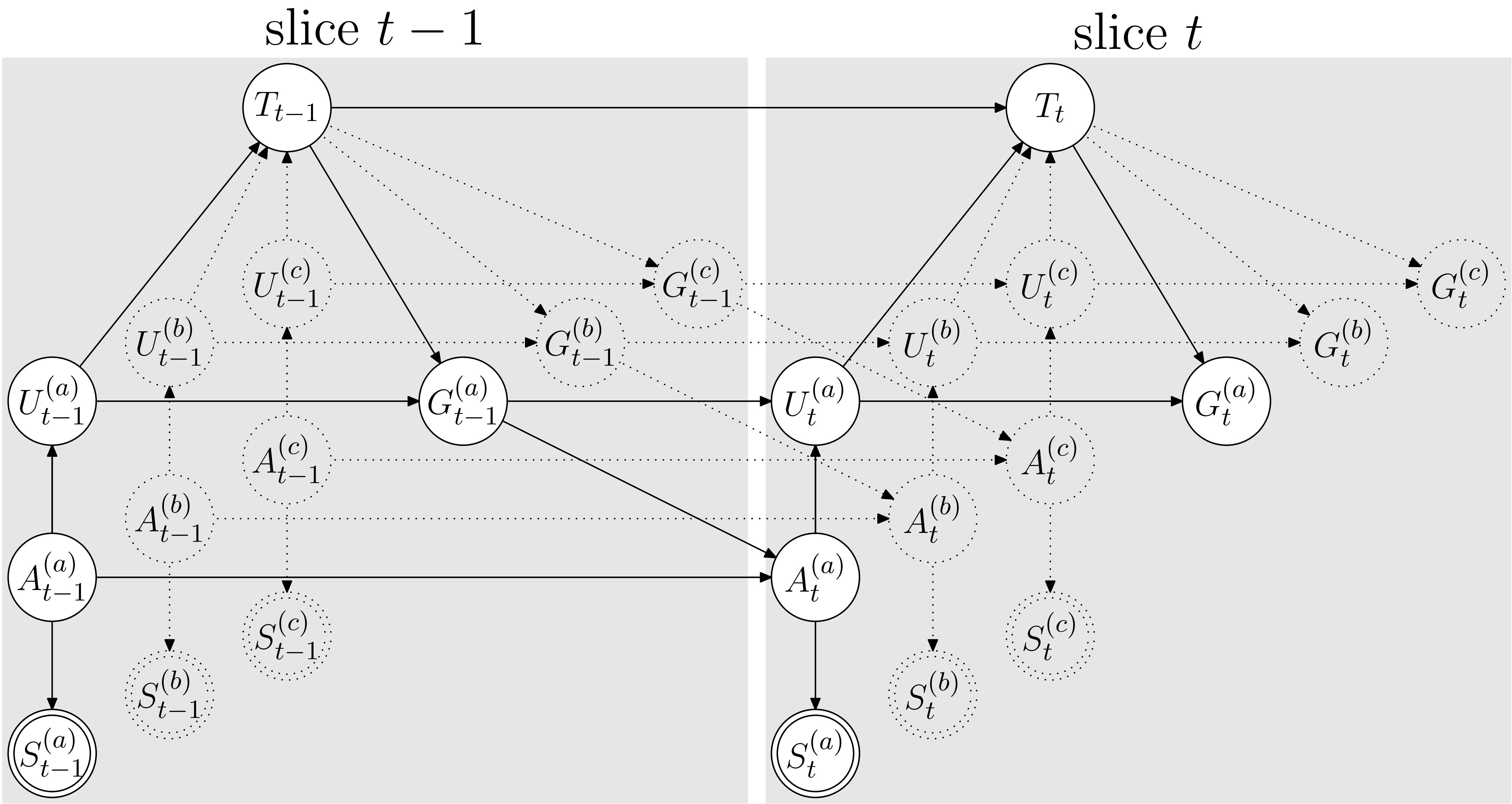


- a) Use Multimodal Interaction
  - this requires the user to interact explicitly
- b) Use Bayesian Inference
  - Build a Dynamic Bayesian Network describing the conditional probability of observations given possible user intentions
    - „If the user wants to present a talk it is likely to observe him in front of the lecture room"
      - (Doing this right isn't trivial)
  - Observe the sensors
  - Then, compute the intention giving the best explanation for the observation
  - This allows to recognize intentions *without* explicit user interaction
    - i.e., without distracting the user from his primary task
  - We have tried to use this approach for recognizing the intention of a *team* of users in a meeting room

Recognizing Team Intentions

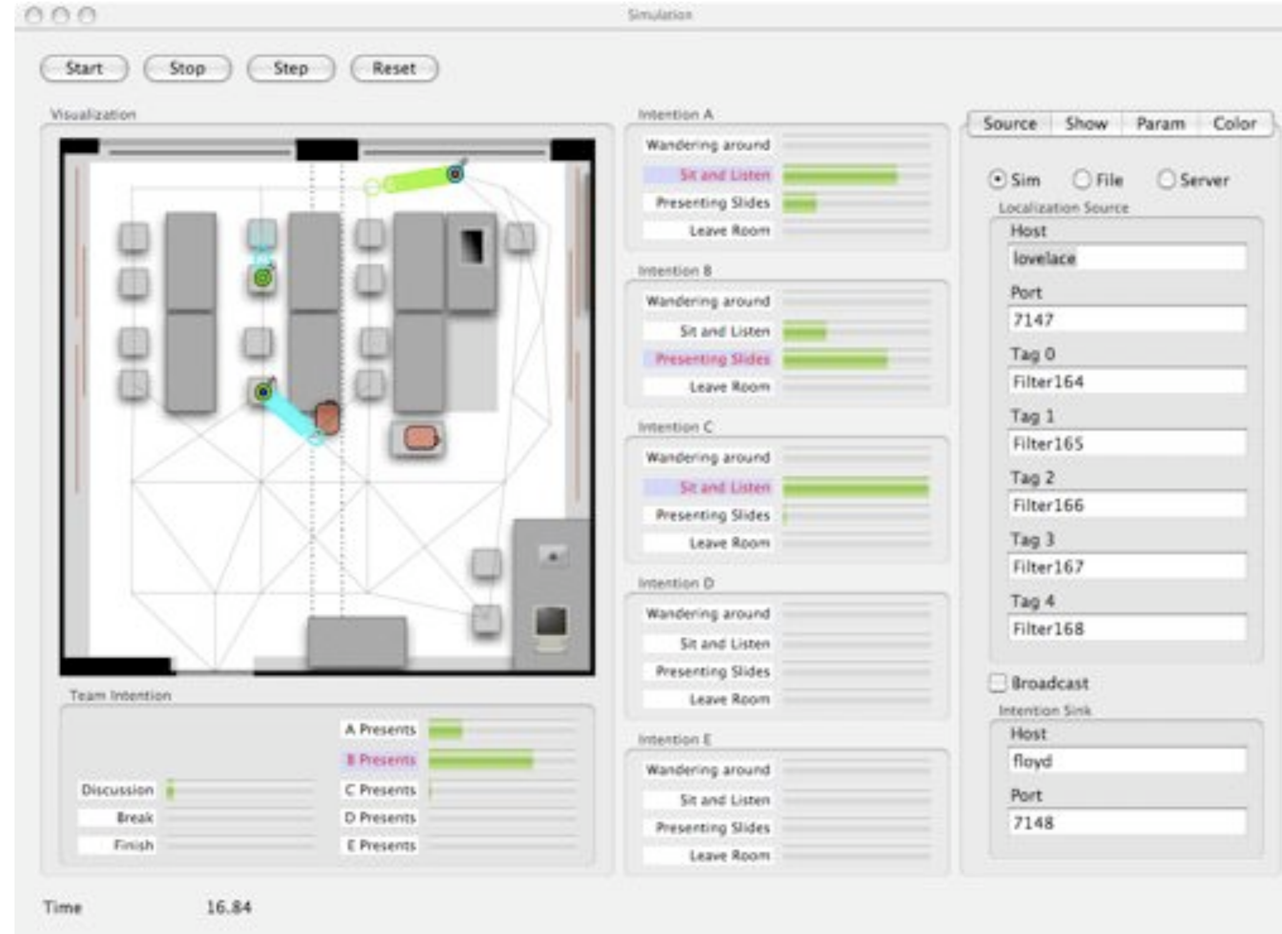
- Agenda provides hints on possible team activity sequence, but can not be enforced
- Questions
  - Can team activities be inferred from position data?
  - When does the system have an estimate of team intentions? (recognition speed / prediction horizon)
  - What role do the PDF parameters play in recognition speed?

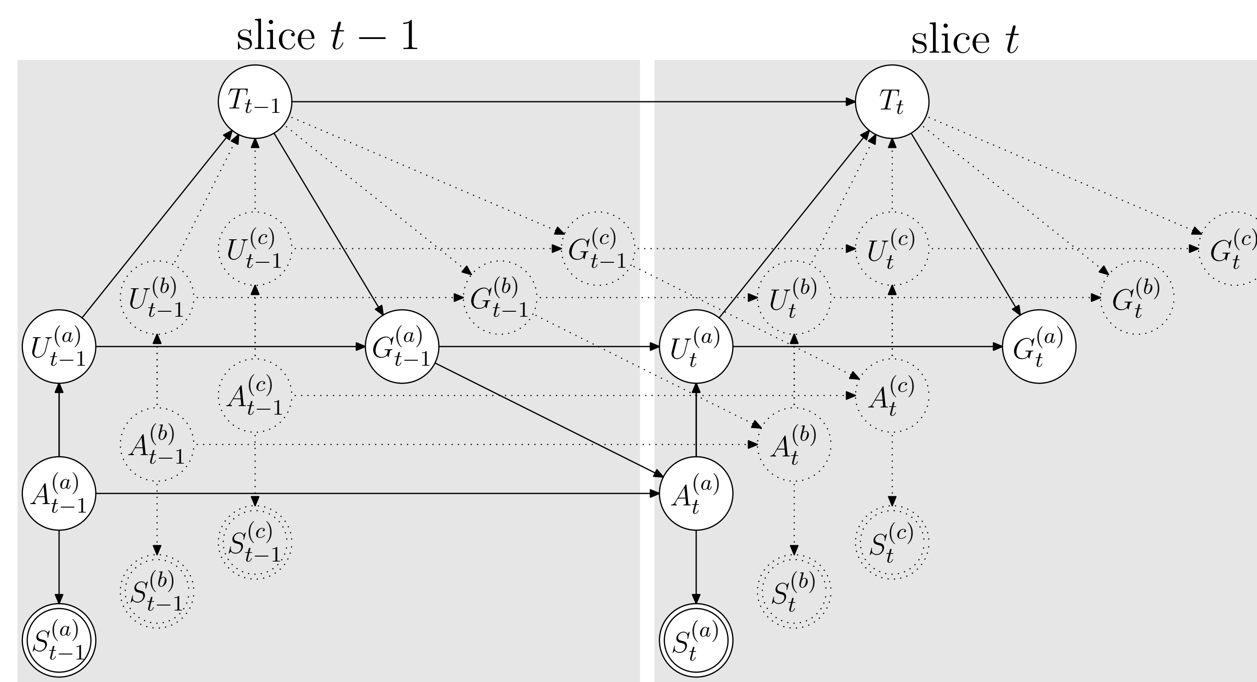
Modeling the Team Decision Process



Model based intention recognition

- Approach
  - Bayesian Tracking
  - Team behavior modeled as DBN
  - Inference using sequential Monte-Carlo
  - Parameter estimation based on EM
- First Results
  - Team behavior can be inferred even with very noisy data
  - Agenda knowledge improves recognition speed , *without* sacrificing performance in case of deviations
  - Parameter estimation (training) essential for recognition speed

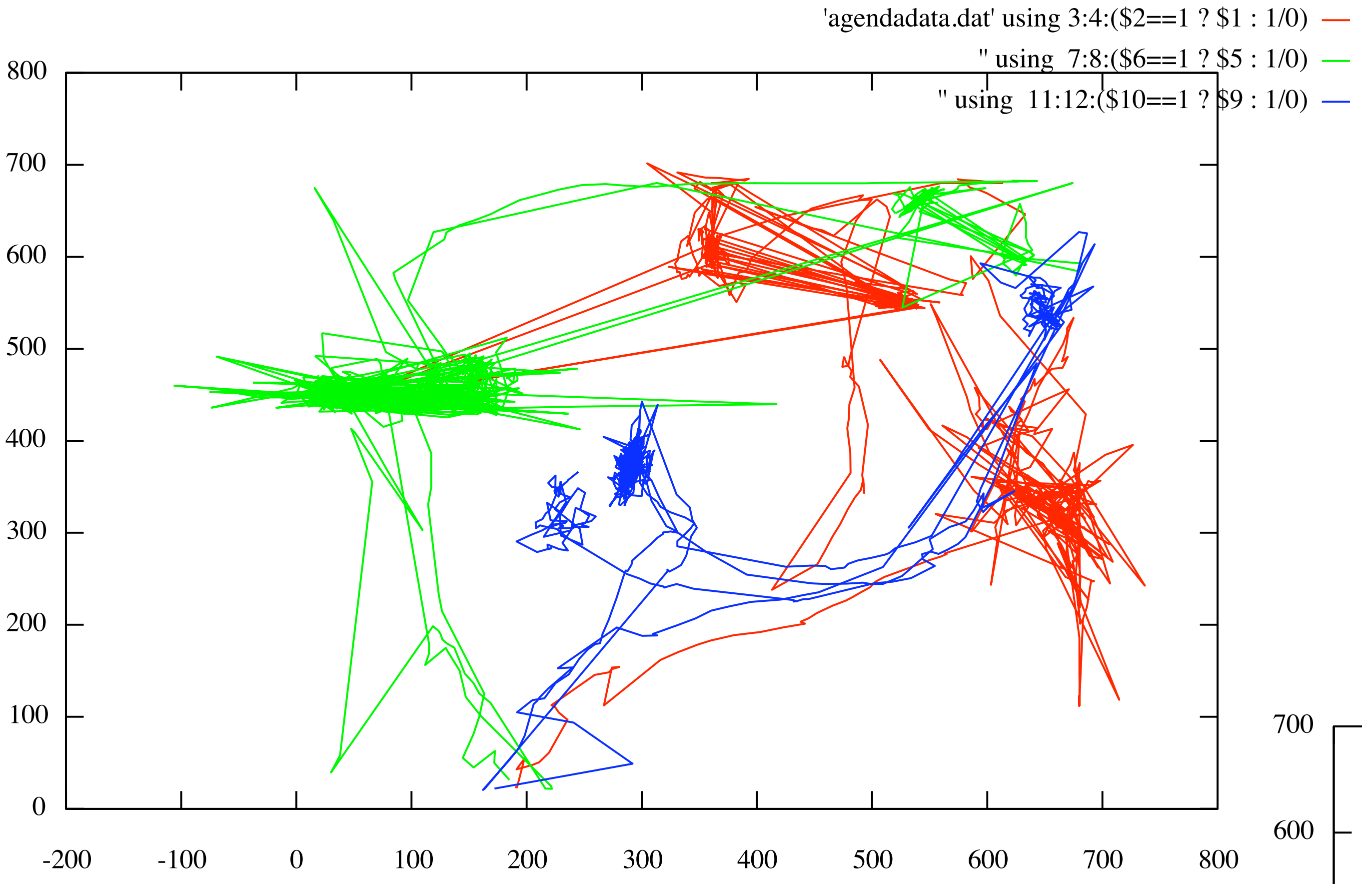




[1] Giersich, M., Forbrig, P., Fuchs, G., Kirste, T., Reichart, D., Schumann, H. Towards an integrated approach for task modeling and human behavior recognition. In *Proc. HCI International 2007: 12th International Conference on Human-Computer Interaction*, Beijing, China, July 22–27 2007.

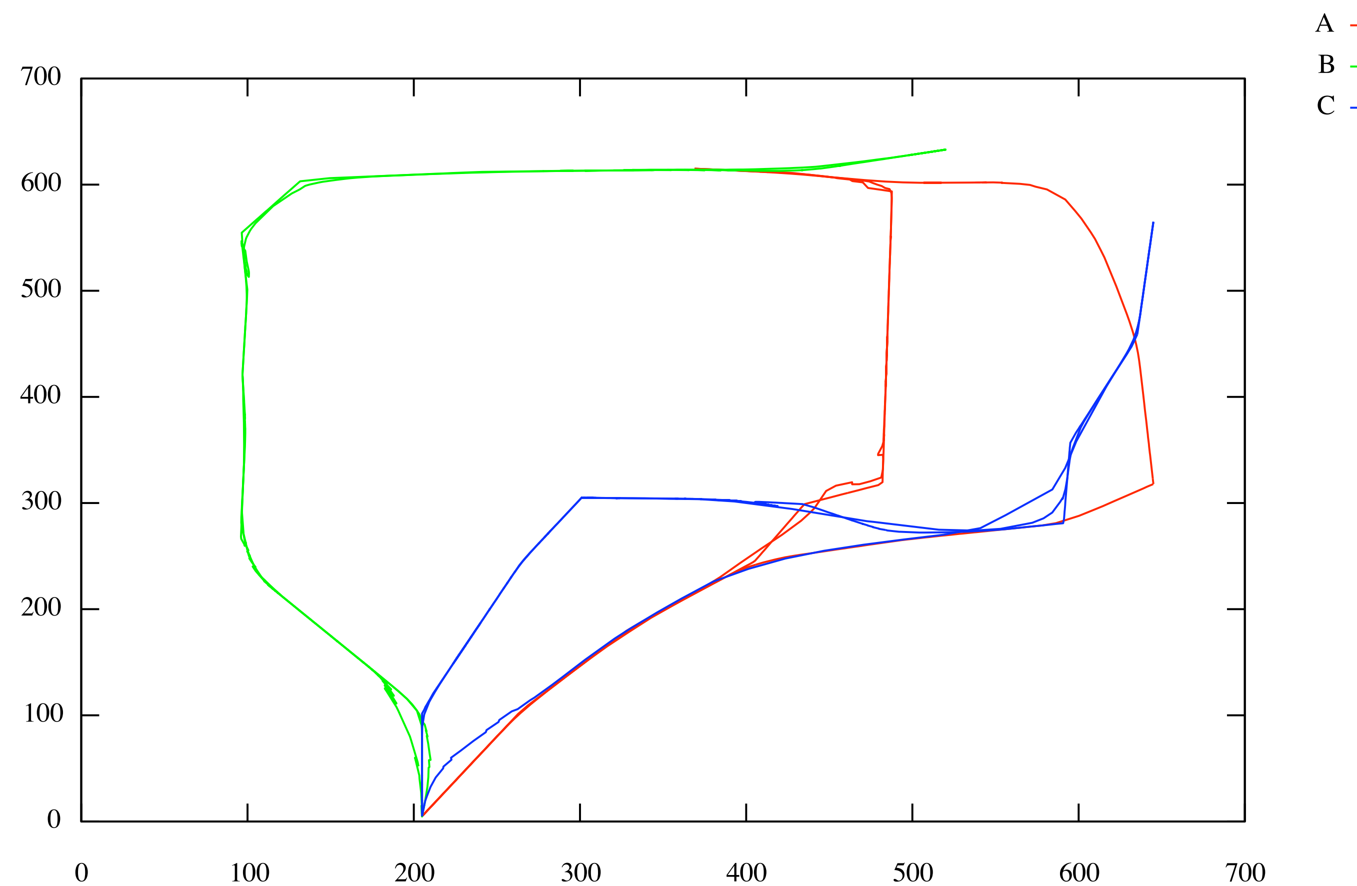
[2] Giersich, M., Reichart, D., Forbrig, P., Kirste, T. Using task models in mobile system design. In *Proc. Dagstuhl Seminar on Software Development Methodologies for Mobile Applications*, Schloss Dagstuhl, Germany, Feb 11–14 2007.

What do we mean by „noisy data“?

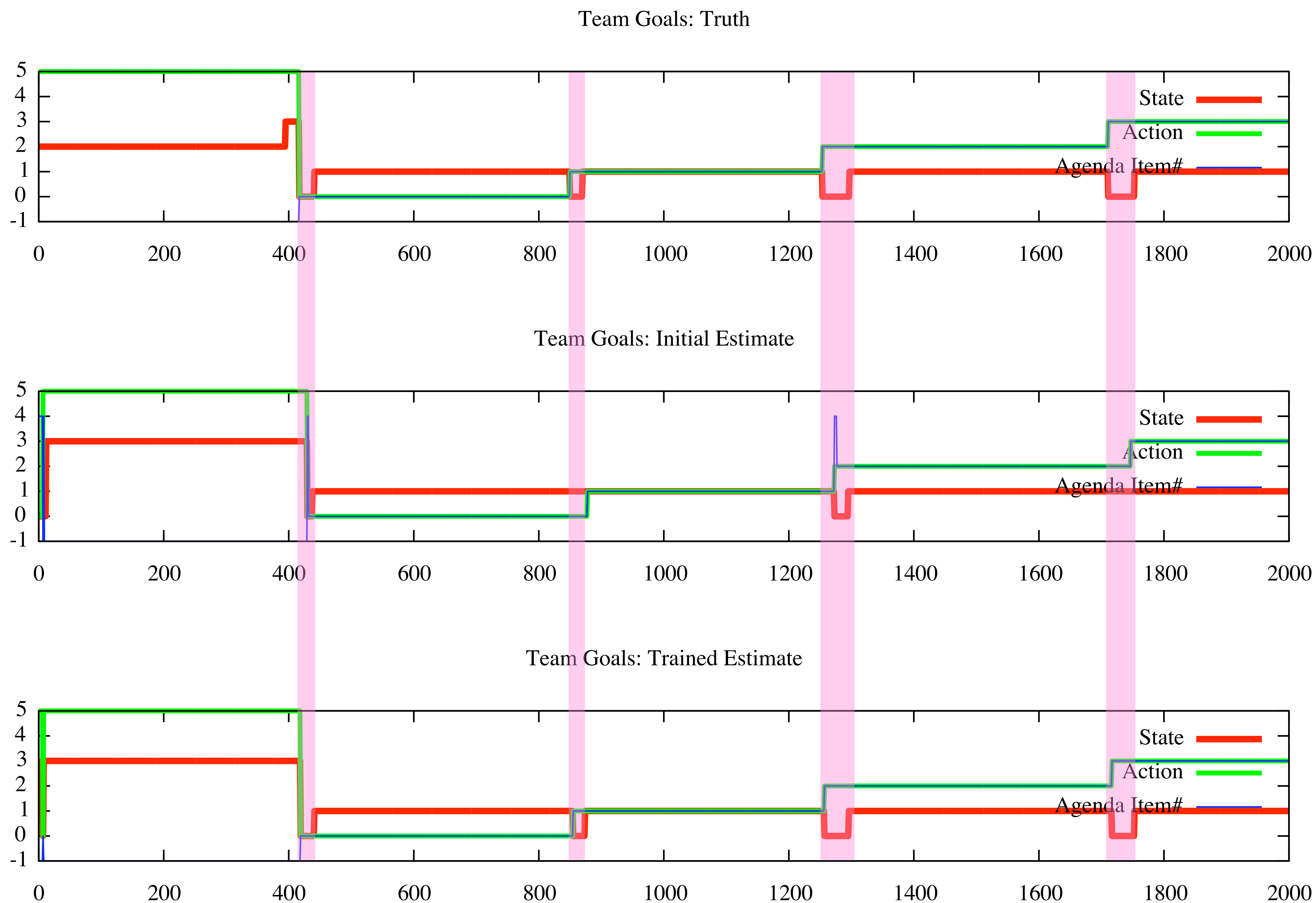


Raw data

Estimated team motion pattern

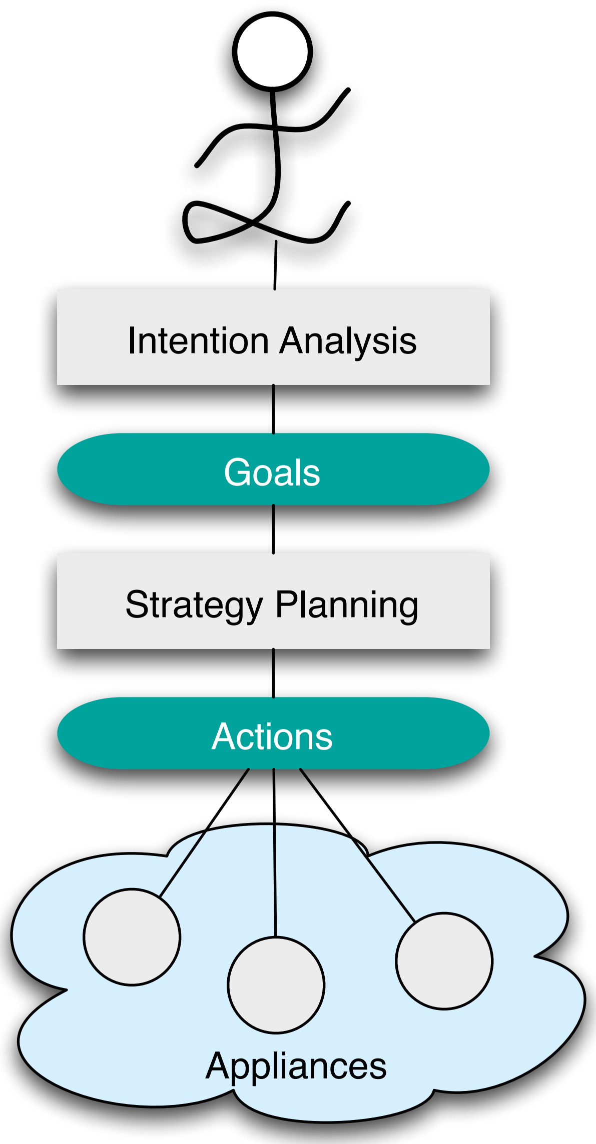


Parameters & Recognition Speed

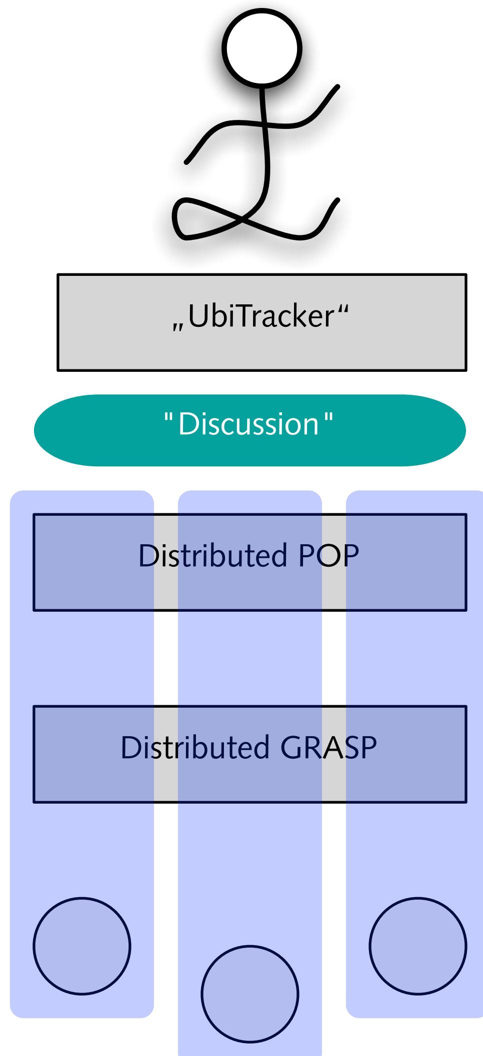


To Summarize …

Concept



Proof of Concept



Bayesian Inference

Action Planning:
Partial Order Planning

Resource Scheduling:
Optimization

Open Questions (Selection)

- Strategy Synthesis
  - Fully distributed („shared nothing“) implementation (with respect to knowing / understanding capabilities of other ensemble members)
    - Efficiency? Naive general algorithms require prohibitive com. overhead
  - Usability? Finding global models that correctly reflect users expectation (Display Mapping) may be a major challenge …
    - Allow to integrate user feedback in sub-optimal models …
  - In general: definition of „universal environment models“
    - Learning? (Problem: Ad-hoc ensembles)
  - Multi-paradigm synthesis. – E.g., integration of
    - reactive approaches
    - Top-Down Planning (e.g., HTN, Service Composition, …)
- Intention Analysis
  - Model-building effort
    - Generate models for available task descriptions (e.g., UI-level task models, CTT)
  - Integration of explicit interaction

Thank you for your attention!

Questions?