

Automatic Device Composition in Service Browsing Environments

Service Browsing Meets Artificial Intelligence

Paul Wisner

Nokia Research Center, Cambridge Mass USA

Paul dot wisner at nokia dot com

Problem Statement

“How can we simplify user interaction with environments that provide a dynamic set of interdependent services.”

- Factors:
 - The available service set is highly dynamic.
 - Configuration of services can be a recurring problem.
 - Configuration of services requires detailed understanding of interactions.
 - Few users know how to do it.
 - Too much effort even for experts.
 - Mobile devices have small screens and limited input.
-

This Presentation Will

- Propose a general scheme for a solution based on
 - A Service Browsing UI Model
 - An Assumption that in the future there will be lots of devices that host IP services
 - Discuss the State of the Art
 - An AI planning approach to service composition
 - Support unanticipated configurations of any number of services
 - Demonstrate AI planning for service composition
 - Propose a User Interface for Nokia S60 phones
-

General Scheme

A Proposal

Paul dot wisner at nokia dot com

The Service Browsing Model

□ Service Browsing

- A desktop continuously scans the environment for all available services. The user selects a service and the system invokes the appropriate client.

□ Differs from Traditional Approach:

- User first selects a client application and then locates compatible services from within that client.
-

Traditional Approach

Launch Control Point application.

User Selects a Media Server



Current Approach

Browsing of Media Server Content.

The image shows a sequence of four overlapping screenshots from a mobile device interface, illustrating the process of browsing media server content. The screenshots are arranged in a cascading manner from top-right to bottom-left.

- Top-right screenshot:** Shows a list of UPnP devices. The first item is "UPnP Control Point" with a mobile phone icon. Below it are "Intel AV Rende..." with IP 172.19.164.49:55217 and "CL2 Media Ser..." with IP 172.19.164.49:58275. An "Exit" button is visible on the right.
- Second screenshot:** Shows the "UPnP Control Point" screen with a list of content items: "3650_snow_brand" (highlighted), "3650_snow_pull", "Bubu Chacha", "Chocolate Energy", "Garfield", and "Only You". At the bottom, there are "Options" and "Back" buttons.
- Third screenshot:** Shows a dialog box with a text field containing the URL "http://172.19.164.49:55217/". Below the field are "Ok" and "Back" buttons.
- Bottom-left screenshot:** Shows the "UPnP Control Point" screen with the same list of content items as the second screenshot.

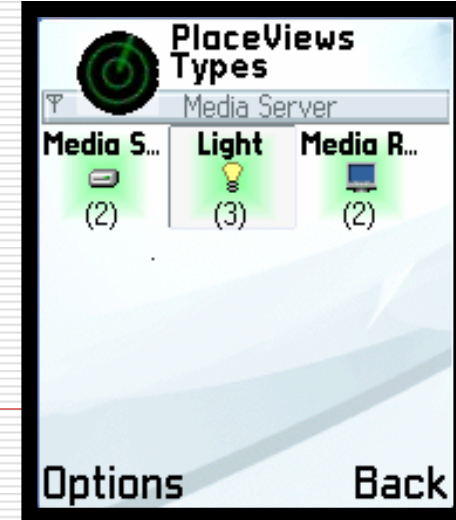
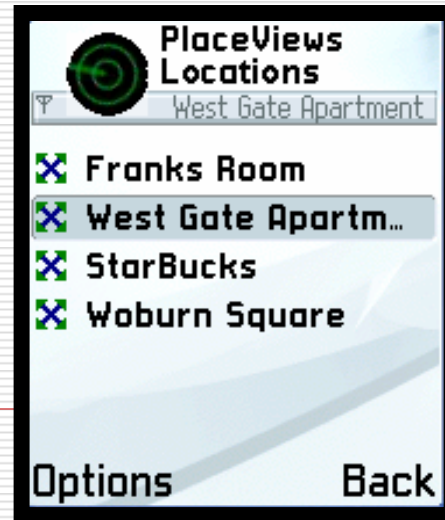
Current Approach



Select a Renderer for Display

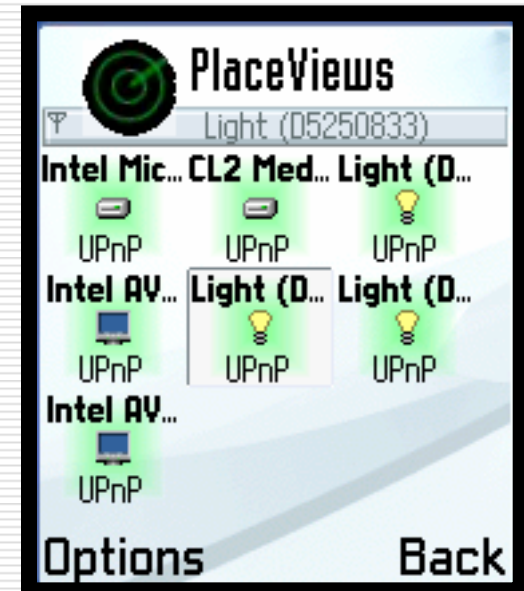
Service Browsing Desktop

- Icons on the screen represent available services.
- Icons appear and disappear as services *come in to* or *go out of "range"*.
 - A service is in "range" if it is reachable and close enough to be useful.
- Service Browsing for
 - Local environments (Smart Spaces)
 - Social overlay networks



Activation


- When user “activates” a service, an “activation module” decides what to do.
 - Table driven. Similar to MIME-TYPES.
 - Example: If Media Server is activated Launch **AV Control Point**



...and Social Overlay Networks

- “Share Your Things with People You Know”
 - Service Browsing: you are now browsing remotely distributed services.
-

Assumptions About the Potential Future

- ❑ Wireless IP connectivity everywhere.
 - ❑ Lots of Devices That Host IP Services
 - ❑ In the future, the device market could change such that there are lots of special purpose devices available in shops.
 - Examples: 
 - ❑ Automatic Composition is the key enabler of this new device market.
-

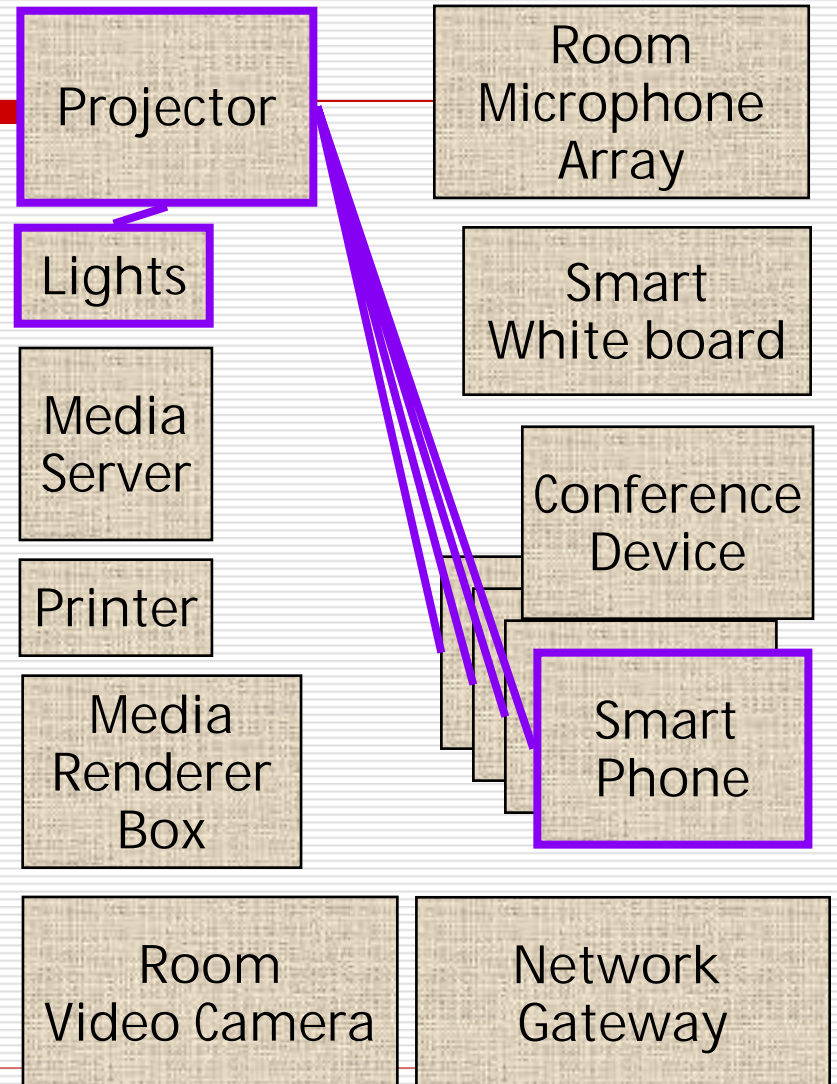
Service Composition

- Service Composition is the process of configuring a set of service providers and service consumers into a useful arrangement.

 - User Interaction:
 - The user selects two or more objects
 - The system proposes ways to use these objects together.
 - The System proposes a Client Application
 - The system selects a client intended for the service
OR
 - The system generates a Situational Application
 - The system can display a representation of the planned composition.
 - Most users will not interact with the composition display
 - Advanced users can inspect and modify the planned composition
-

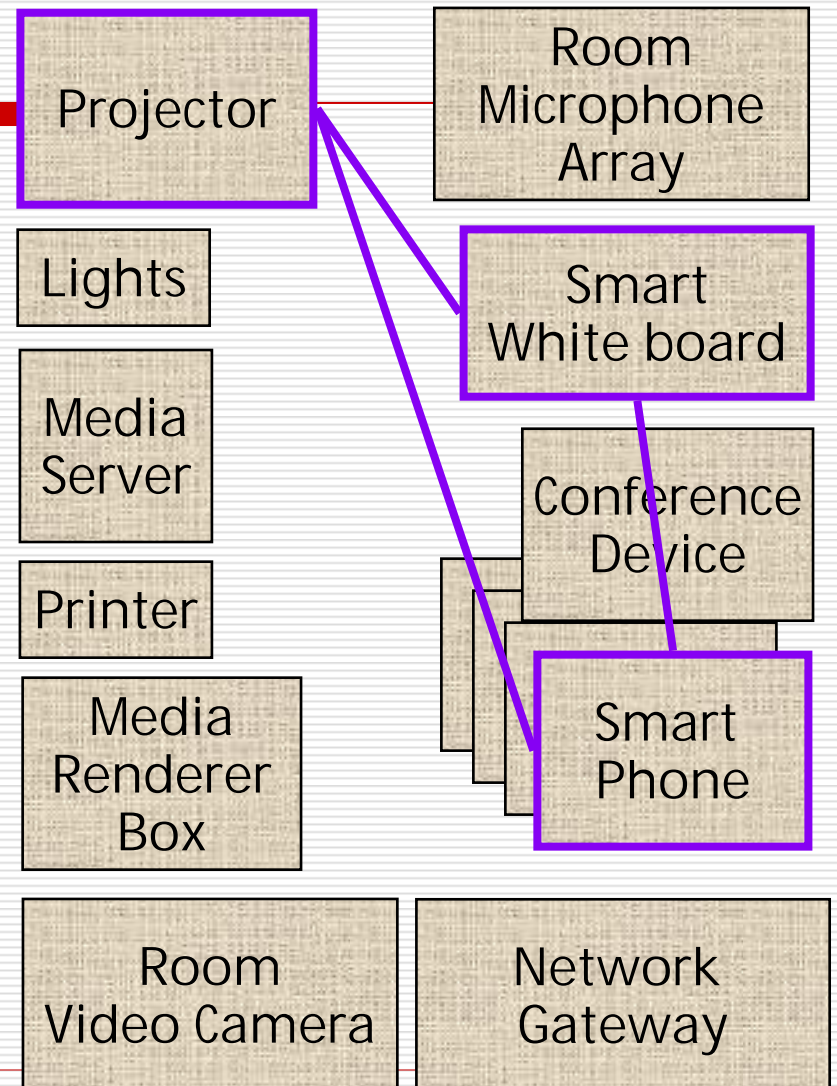
Use Case 1 – Projector + Smart Phone

- Smart Phone can
 - display a document in it's memory on the Projector.
 - control scrolling.
 - make annotations.
- Lights are dimmed.
- Multiple Smart Phones can
 - Share the display – split view for multiple documents.
 - Share control – multiple clients making annotations and controlling scrolling.



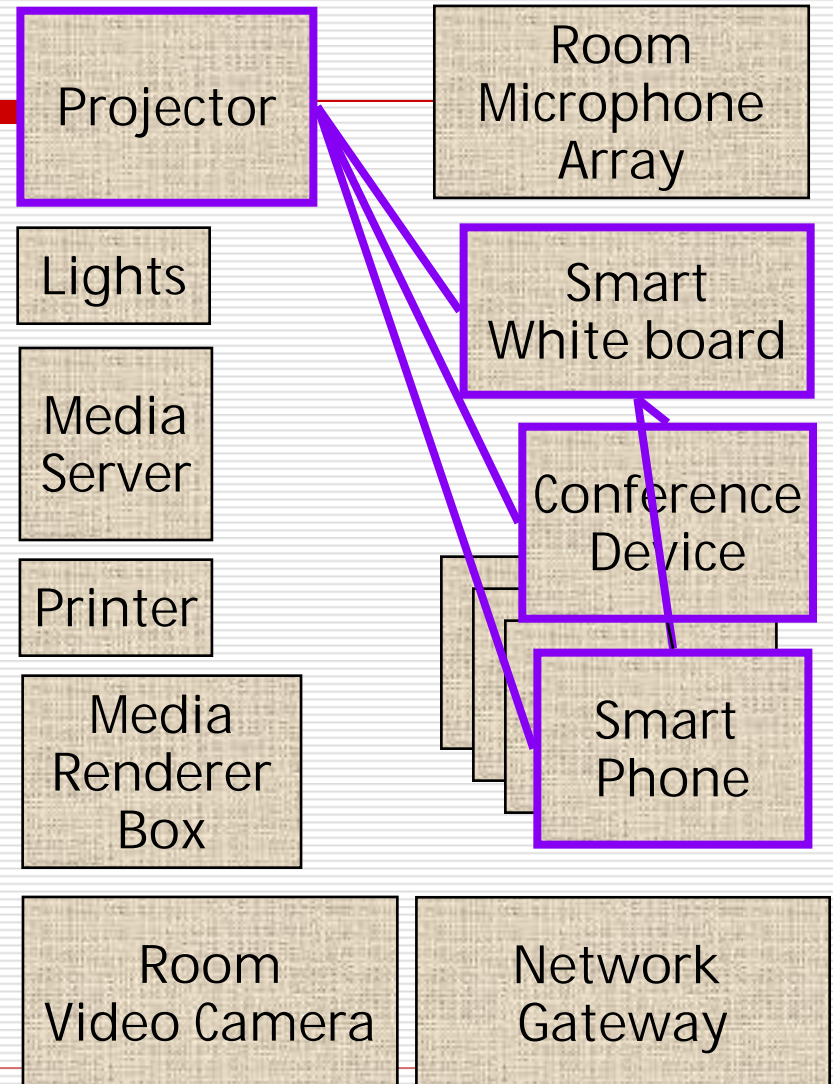
Use Case 2 – Projector + Smart Phone + Smart Whiteboard

- User indicates that he wants to use the whiteboard to annotate the slides.
 - The projector aims it's image onto the whiteboard.
 - User can now draw in the projected image. The user's drawing actions modify the document being displayed.



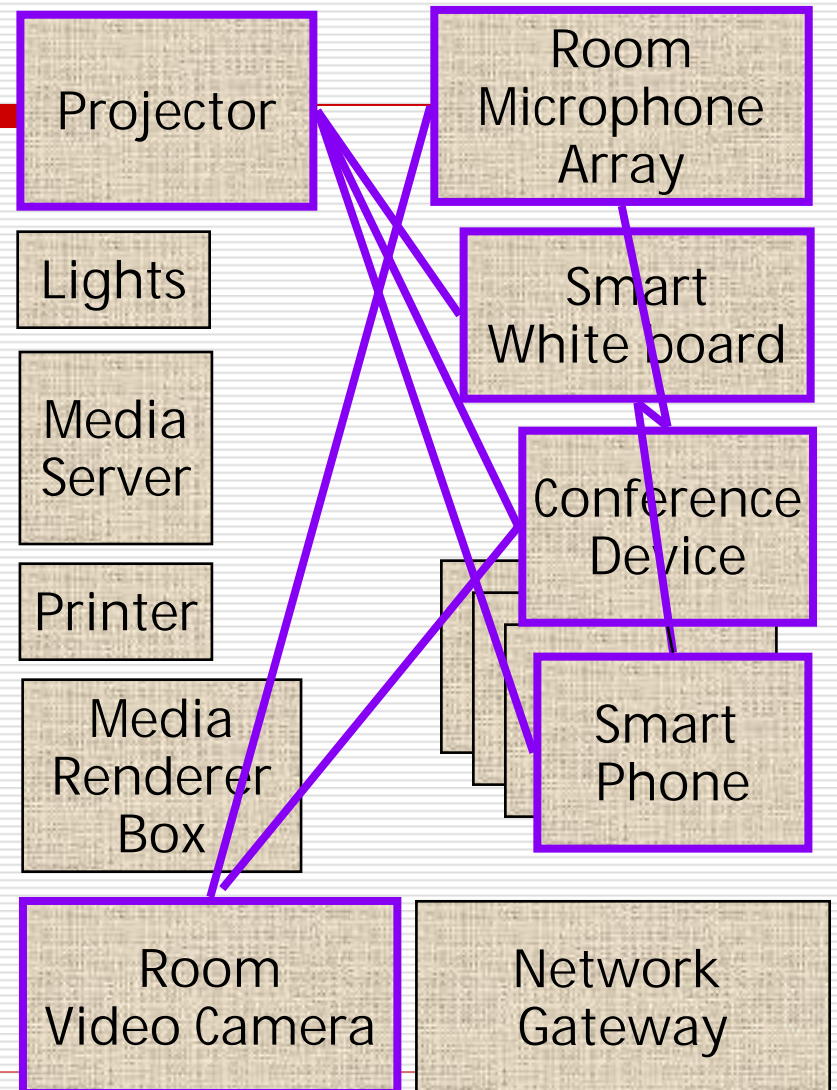
Use Case 3 – Conferencing + Projector + Smart Phone + Smart Whiteboard

- User decides to invite a remote party to the meeting.
 - The Conferencing Device is used to share the media experience with the remote party.
 - The remote party is also visible on the projector.



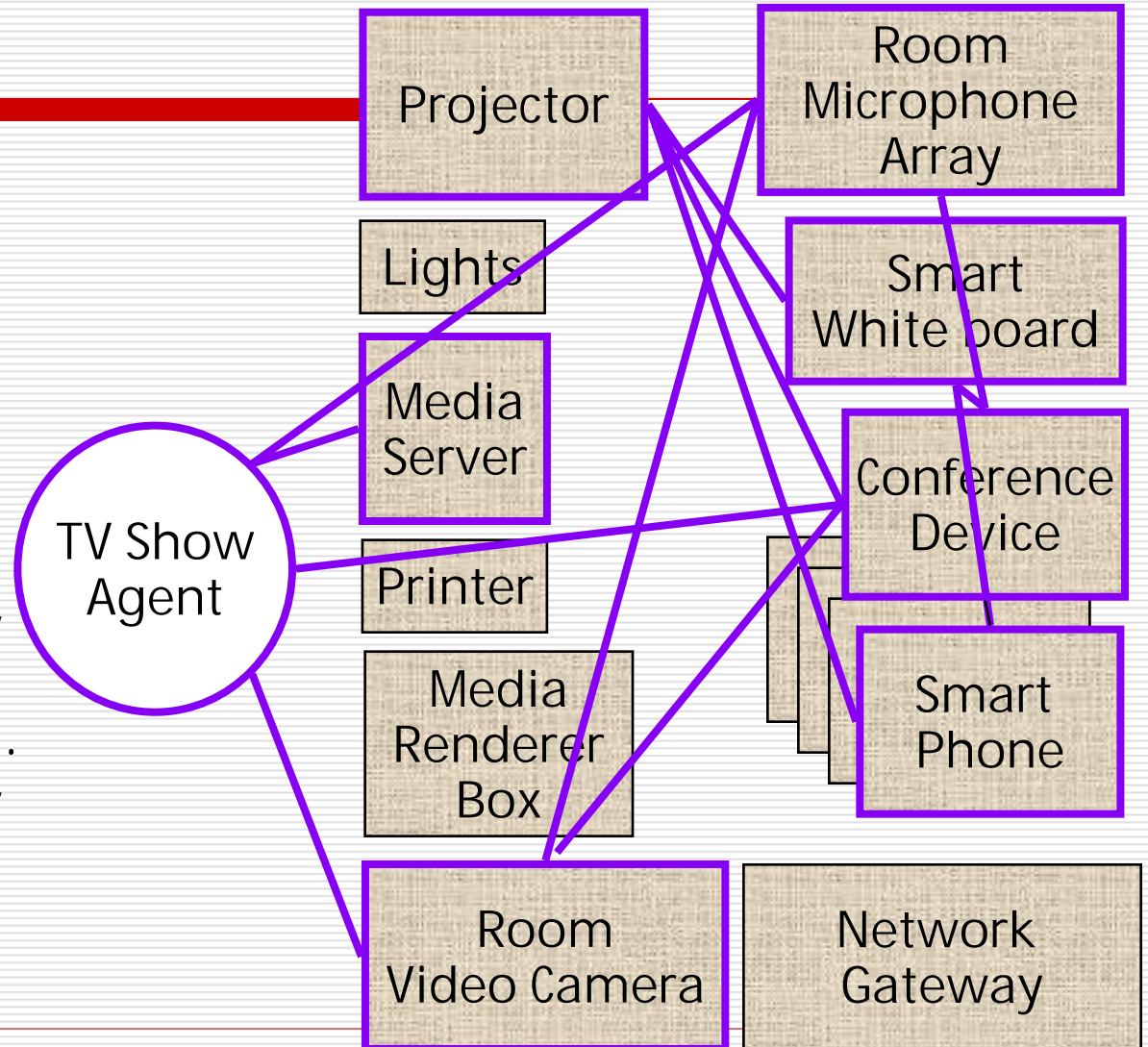
Use Case 4 – Conferencing + Room Video + Projector + Smart Phone + Smart Whiteboard

- But the remote party does not really feel involved in the meeting so he requests video.
- The Room Video Camera joins the system.
- The Video Camera uses the Room Microphone Array to aim at whoever is speaking or to Zoom out to a wide view when there are many speakers.



Use Case 5 – Meeting TV show

- A TV Show Agent creates a TV Show of the meeting and records to the Media Server.
 - Most active periods are highlighted.
 - Automatically show whiteboard when drawing is occurring.
 - Automatically show speaker.



State of the Art

- Pre-Defined Collaboration
 - Two devices are designed to work together for a single purpose.
 - Task Computing
 - Key Difference: Assumes a set of “known” tasks has been anticipated.
 - Lots of “Service Composition” Research from Web Services and Semantic Web
 - Good results for representation
 - Some are too complex or computationally difficult: OWL-S
 - Mostly Relevant
 - But, some different assumptions:
 - Fixed network
 - Static service availability
 - Established AI techniques
 - Inference engines
 - Planning Systems
 - Agent paradigm
 - Distributed Intelligence
-

Self Describing Interfaces

□ Basic: interface name

Assumes the author of the client application understands the interface

- Java RMI, Jini, Bonjour
- IDL descriptions only available at compile time
 - Corba, SunRPC, ActiveX, RPC

□ Basic: methods and arguments

The client can determine the methods and their arguments at run-time

- Java introspection
- SOAP, XML-RPC, UPnP, WS: XML based

□ Advanced: semantics

Encoding what the interface does and how it is used.

- OWL-S
 - Semantic Web Services
-

Our Service Composition Engine

- ❑ An Early Implementation
 - ❑ Based on interface name matching
 - ❑ Demonstrates Feasibility of Service Composition using simple interface description and established AI techniques.
 - ❑ Centralize planning engine.
-

The Device Composition Aquarium

- A Workbench for experimenting with Knowledge Representation and Inferencing Schemes.
 - Assumes Devices host IP Services
 - Demo
-

Weaknesses of current implementation

- Poor estimation of optimality
 - Seems to require domain specific heuristics

 - Space complexity too great
 - Need heuristic function to provide “score” of each search path.

 - Central Planning
 - Where is the planning engine hosted?
-

Ideal Composition Engine

- Good Heuristic Function
 - Provides estimation of optimality to guide search and narrow search space
 - Based on domain knowledge
 - Based on current context
 - Distributed Planning Engine
 - Each participant plans for its immediate needs
 - Supports totally adhoc scenarios: does not assume intelligent infrastructure
 - Relevant work
 - MAP -- Multi Agent Planning
 - DCOP, DPOP, ADAPT - Distributed Constraint Based Planning
-

But How Would This Look to the User?

- Start with Service Browsing
 - Select some “objects” to use together
 - System generates a plan and proposes some client applications
 - Two Approaches to client applications
 - Short term: Select a known client application
 - Long term: Generate a “Situational Application”
 - Make the proposed composition plan available for inspection.
-

Screen Shots Mockups

Nothing yet....!

Select a known client application

- ❑ Short Term Client Application Approach
 - ❑ Table Driven application selection
 - Similar to web browser MIME-TYPE
 - Also make downloadable clients available
-

Generate a “Situational Application”

- Long Term Client Application Approach
 - Based on Self Describing Interfaces
 - User functions are generated based on methods listed in the interface description
 - These functions become menu items and buttons
 - Dialog boxes are generated for filling in arguments
 - Using Semantic Descriptions
 - How can we utilize richer descriptions?
-

Screen Shot Mockups

Nothing yet...!

Composition Plan Inspection Screen

- This is a screen that helps the user understand how devices are interacting.
 - This will also provide a UI to enable the user to override specific decisions made by the device composition engine.
-

Summary

- Combining Service Browsing Model with Automatic Composition enables sophisticated usage by untrained users.
 - Can be accomplished by combining existing technologies
 - Next steps:
 - Built it
 - Distributed planning
 - Heuristics for optimality
-