

1

Introdução

Avanços computacionais evidentes, tais como maior poder de processamento e armazenamento, miniaturização, interconectividade, maior capacidade de baterias, redes sem fio e tecnologias de software, vêm tornando o computador cada vez mais integrado à vida cotidiana, e um modelo a que chamamos *computação ubíqua*¹ vem sendo vislumbrado há alguns anos [4, 6, 10, 16, 26, 27].

Cada vez mais freqüentemente ambientes físicos estão sendo dotados de muitos e variados dispositivos computacionais que aumentam a percepção e a imersão do usuário com informações e processamento digital. Esses novos ambientes de trabalho têm como objetivo disponibilizar uma grande quantidade de dispositivos computacionais que ofereçam novas funcionalidades, aumentem a produtividade dos usuários e facilitem as tarefas do cotidiano. Em casa, no trabalho e em áreas públicas, uma infra-estrutura computacional ubíqua é capaz de otimizar o ambiente do usuário para atender às necessidades de suas atividades de trabalho e recreativas. Os usuários podem ter acesso em qualquer lugar e a qualquer instante à rede e a recursos computacionais.

Neste novo contexto, as aplicações devem fazer uso efetivo dos recursos disponíveis para dar suporte às atividades dos usuários. Entretanto, a heterogeneidade, a mobilidade e o grande número de dispositivos tornam o desenvolvimento de aplicações uma tarefa muito mais complexa. Com o intuito de facilitar esse desenvolvimento, foi definido o conceito de *espaço ativo* [26], que é um espaço físico estendido por uma série de dispositivos computacionais e uma infra-estrutura de software responsável por gerenciar os recursos disponíveis e coordenar as atividades que se desenvolvem nesse espaço. Essa infra-estrutura de software desempenha um papel similar ao de um sistema operacional em um computador convencional, definindo

¹O termo *computação ubíqua* vem do inglês *ubiquitous computing* e conota onipresença, mas já ganhou outros nomes como *pervasive computing*, referente à sua característica invasiva, e *seamless computing*, em tradução literal *computação sem emendas*, relativo ao seu aspecto de integração de sistemas.

um ambiente computacional genérico que converte um espaço físico e seus dispositivos em um sistema programável.

Um exemplo de espaço físico que carece de tal infra-estrutura de software e que vem se tornando cada vez mais comum são as *salas de trabalho multifuncionais*, ambientes de trabalho que contam com uma série de recursos computacionais e de multimídia para dar apoio a atividades como reuniões, apresentações e sessões de trabalho colaborativo. Essas atividades podem se desenvolver com todos os participantes presentes no mesmo ambiente de trabalho, com alguns participantes remotos ou até mesmo com participantes distribuídos por salas em diferentes localidades.

Diversos trabalhos [26, 27, 44] oferecem uma infra-estrutura genérica para o desenvolvimento de aplicações para espaços ativos. O projeto Gaia [26], desenvolvido pelo Departamento de Ciência da Computação da Universidade de Illinois em Urbana-Champaign, é um meta-sistema operacional que objetiva a criação de um novo nível de abstração para a operação e o controle de múltiplos dispositivos e aplicações. Tal infra-estrutura de software visa solucionar diversas questões comuns ao uso de ambientes de computação ubíqua. Início e término remoto de aplicações, integração entre diferentes middlewares [9, 17], operações sobre o sistema de arquivos [39], detecção de erros, segurança [28] e alocação de recursos são alguns dos serviços que se fazem necessários em tais ambientes.

Além disso, o projeto Gaia vem estudando a criação de novos tipos de aplicações para a realização de diferentes tarefas sobre tal infra-estrutura, como por exemplo trabalho colaborativo, antecipação e adaptação da execução de aplicações, obtenção e recuperação do contexto de execução [45], adaptação de execução de aplicações entre salas [55], integração de aplicações *single-user* para controle remoto [46] e a integração de diferentes redes de espaços ativos criando a noção de *superspaces*.

Outra aplicação possível é a utilização de tais ambientes ricos em dispositivos para a execução coordenada de apresentações multimídia compostas por vídeo, som, imagens estáticas, páginas web ou até mesmo aplicações convencionais que não estejam diretamente ligadas a conteúdo multimídia. O uso de múltiplos dispositivos e aplicações para compor uma apresentação distribuída oferece diversos desafios, mas ao mesmo tempo oferece grandes possibilidades. A combinação de diversas telas em uma apresentação normalmente causa um grande impacto na audiência. Mais ainda, dispositivos para a captura de informações sobre o espaço físico, tais como os Ubisensor [1] e MotionStar [2], câmeras que podem ser remotamente controladas, telefones celulares e PDAs podem aumentar a interatividade

de uma apresentação. Finalmente, aplicações *single-user* já estabelecidas, como Microsoft PowerPoint, Windows Media Player e Internet Explorer, facilitam ao usuário a criação de conteúdo para apresentações.

A variedade de recursos disponíveis em salas multifuncionais permite que o mesmo conteúdo seja apresentado em diferentes dispositivos. De forma similar, tal conteúdo pode ser apresentado simultaneamente em dispositivos de outros espaços ativos, ou até mesmo no computador de um participante remoto. Esse contexto de trabalho naturalmente propicia o desenvolvimento de atividades colaborativas, nas quais vários usuários interagem simultaneamente com o ambiente, sejam esses usuários locais ou remotos. Desta forma, a distinção entre a realização de um trabalho colaborativo e de uma apresentação multimídia se torna bastante difícil.

Como exemplo, considere um cenário de uma sala multifuncional com múltiplas telas, cada uma acoplada a um computador diferente. Uma apresentação multimídia pode ser construída de forma a utilizar duas telas para a reprodução de um vídeo de forma sincronizada. Isto permite ao espectador olhar para qualquer tela para acompanhar o vídeo, o que além de trazer maior conforto também cria um efeito de imersão na apresentação. Em um terceiro computador, *slides* PowerPoint® podem ser apresentados fazendo referência ao conteúdo do vídeo. À medida que diferentes tópicos forem abordados no vídeo, os slides são avançados automaticamente. Um outro computador pode executar um navegador web, também sincronizado com o vídeo, de forma a mostrar páginas que contêm tópicos relativos ao conteúdo do vídeo.

Em um segundo cenário, considere uma apresentação multimídia distribuída onde alunos assistem a uma aula através de seus espaços ativos. Em uma tela podem ser apresentados *slides* PowerPoint® relativos à aula. Uma segunda tela pode mostrar uma aplicação que recebe mensagens instantâneas para onde são enviadas mensagens de texto relativas ao conteúdo de cada *slide*. O texto enviado também pode ser lido através de uma aplicação de TTS (*Text To Speech*). Cada vez que o aluno recebe uma mensagem a aula prossegue, avançando o *slide* PowerPoint. Além disso, em determinados pontos da apresentação, vídeos que enfatizam o conteúdo exposto podem ser iniciados e terminados em telas auxiliares.

Em um cenário diferente desejamos controlar diversos dispositivos em uma peça teatral. Nesta situação, computadores poderiam controlar som, vídeo, iluminação, entre outros efeitos durante a realização de uma peça. Além disso, dispositivos de rastreamento da posição dos atores [1, 2, 18] podem auxiliar a coordenação da performance, criando efeitos baseados em

suas posições. Por exemplo, o controle do volume do som pode ser ligado à posição de um ator. Tal efeito poderia ser utilizado para interromper uma música quando o ator deixa o palco. Em outro exemplo, a posição dos atores poderia guiar uma luz direcional para sua atual posição ou até mesmo iniciar a reprodução de um vídeo.

Como exemplos de projetos atuais que poderiam fazer uso de uma infra-estrutura de controle de apresentações multimídia distribuídas, temos o CyberOpera e as obras de arte de Camille Utterback [3]. O projeto CyberOpera, que vem sendo desenvolvido pela Universidade de Illinois em Urbana-Champaign, tem como objetivo a construção de uma peça teatral controlada por computadores. Este projeto piloto e seus resultados servirão para direcionar a criação de um novo modelo de teatro automatizado. O Krannert Center e o Departamento de Computação desta universidade vêm buscando patrocínio para a construção de um prédio com este objetivo.

Finalmente, o trabalho de Camille Utterback vem explorando novas tecnologias de computação para a criação de obras de arte através da geração de efeitos visuais e sonoros com base na movimentação de pessoas e objetos em um espaço físico. Galerias de arte, museus de ciência e museus para crianças vêm se interessando por este tipo de trabalho tanto como forma de arte quanto de entretenimento. Como exemplos dos efeitos criados através da posição de espectadores, temos diversas brincadeiras com linhas e cores, a variação da velocidade de reprodução de segmentos de um vídeo, uso de projeções no piso para a criação de efeitos, entre outros.

1.1

Objetivos e Contribuições

O objetivo principal deste trabalho é a criação de um sistema capaz de representar e executar apresentações multimídia distribuídas. Para isso, experimentamos a construção de um sistema de execução de apresentações. Além disso, experimentamos o uso de uma abstração de representação de apresentações através de documentos multimídia.

Para definir quais requisitos gostaríamos de abordar no sistema de execução dessas apresentações, imaginamos cenários e construímos diversos protótipos para indicar quais seriam os serviços e componentes de software necessários. Nota-se contudo que, em um sistema de computação ubíqua, diversos componentes já estão disponíveis para utilização. Nos próximos capítulos veremos quais componentes de software são necessários e como a solução apresentada faz uso deles.

Nossos protótipos também serviram para guiar os requisitos de representação da apresentação e para guiar uma posterior construção do software de coordenação da apresentação. Nosso objetivo é utilizar um documento multimídia que permita ao usuário descrever as aplicações a serem utilizadas, o conteúdo de cada aplicação (a mídia) e a ligação entre cada aplicação. Um dos benefícios dessa estratégia é o desacoplamento do desenvolvimento de duas partes independentes: a plataforma de execução e o sistema de autoria do documento. O segundo deve auxiliar usuários menos experientes na construção de suas apresentações distribuídas.

Uma das características que gostaríamos de obter é flexibilidade para inclusão de novas aplicações em nosso sistema de execução. Isso permite que novos tipos de mídia e conteúdos possam ser apresentados, oferecendo extensibilidade ao sistema. Mais ainda, a inclusão de novas aplicações provê ao sistema liberdade para a criação de apresentações multimídia através de aplicações que não foram criadas para este fim, como por exemplo, uma aplicação de renderização tridimensional em tempo real.

Além disso, diferentes aplicações oferecem uma variedade de mecanismos de entrada de dados e comandos, além da possibilidade de múltiplas interfaces de entrada como teclados, *mouse*, voz e gestos. Da mesma forma, as aplicações podem ter à sua disposição uma série de dispositivos de saída, tais como monitores de vídeo, telas de PDAs, projetores, telões e alto-falantes. Dispositivos e componentes de software também podem ser adicionados ou removidos do ambiente a qualquer instante, e assim as aplicações precisam ser capazes de se adaptar dinamicamente a novos ambientes de execução. Muitos aspectos do contexto físico, computacional e comportamental que circundam o usuário podem ter impacto no comportamento da apresentação, devido à disponibilidade de recursos, segurança e privacidade. Através do controle das aplicações é possível relacionar futuros requisitos, dispositivos e tecnologias à reprodução de uma apresentação.

Além da infra-estrutura de execução, é necessário um sistema para guiar tais apresentações multimídias. Atualmente muitas aplicações usam formatos de documentos para descrever apresentações em uma máquina [19, 11, 20, 47, 48]. Apesar de alguns desses trabalhos servirem de base para a representação de um documento, nota-se uma diferença no objetivo final de cada formato bem como no uso das aplicações, como é descrito no Capítulo 7.

Outro obstáculo na descrição do documento é como flexibilizá-lo para permitir a inclusão de novas aplicações. Cada aplicação possui características e interface de acesso completamente diferentes. Além disso,

a relação entre cada aplicação (e sua forma de controle) precisa também ser representada. Idealmente deseja-se que não seja necessário alterar o formato cada vez que uma nova aplicação precisa ser adicionada.

Gostaríamos também que a descrição da apresentação seja independente do espaço físico onde ela é executada. Desta forma, é possível que o sistema tenha informações suficientes para adaptar automaticamente um documento entre diferentes salas multifuncionais. Para tanto, faz-se necessária a generalização dos dispositivos e das aplicações.

Uma das principais contribuições deste trabalho é a construção de um sistema para execução de apresentações multimídia distribuídas, utilizando conceitos e resultados obtidos através da pesquisa na área de computação ubíqua. Além dessa, outra contribuição importante é a especificação de um formato de documento capaz de representar e guiar apresentações em ambientes de computação ubíqua. Ambas contribuições foram testadas e verificadas através dos cenários e protótipos apresentados ao longo deste trabalho.

Esta dissertação está organizada da seguinte forma: no Capítulo 2, analisamos algumas características e desafios para o desenvolvimento de aplicações para salas multifuncionais sob o ponto de vista de *espaços ativos* [26], comentando e comparando seus objetivos e soluções; o Capítulo 3 apresenta protótipos de aplicações construídas com esta infra-estrutura; descrevemos no Capítulo 4 a infra-estrutura de execução para a construção de apresentações multimídia distribuídas, apresentando algumas de nossas decisões de projeto e a arquitetura de nossa solução; o Capítulo 5 apresenta SMIL, uma linguagem bastante utilizada para a representação de apresentações multimídia, e NCL, uma linguagem com maior poder de representação que por fim optamos por utilizar. Neste capítulo também propomos modificações à linguagem NCL para adequá-la a ambientes de computação ubíqua, e apresentamos exemplos de documentos. No Capítulo 7 apresentamos trabalhos relacionados. Finalmente, o Capítulo 8 destina-se às conclusões desta dissertação e trabalhos futuros.