PSL CHIME & AI²TV

Immersive 3D Collaboration Environments

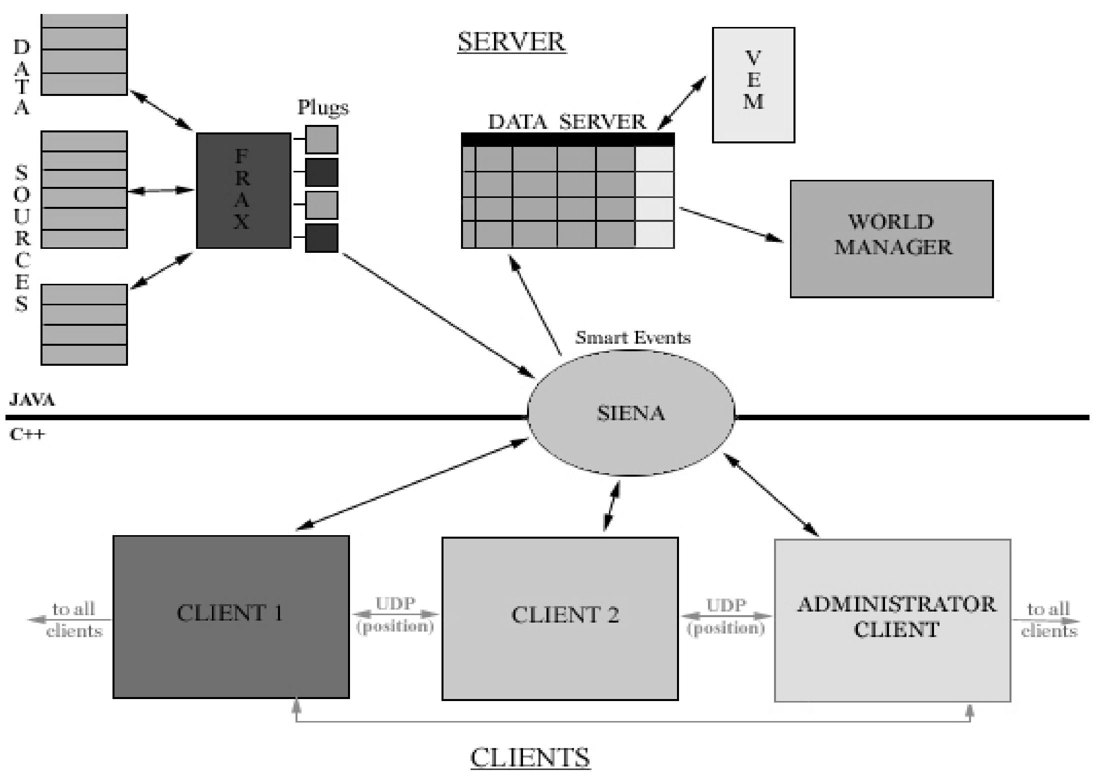
PROGRAMMING SYSTEMS LAB, COLUMBIA UNIVERSITY

CHIME

Introduction

The ***CHIME*** (**C**olumbia **H**ypermedia **IM**mersion **E**nvironment) 3D virtual environment framework supports multi-party/multi-team collaborations as well as individual knowledge workers and decision makers. Users see and touch project artifacts while "walking around" a virtual world, as in games like Unreal Tournament and Quake. MUD (Multi-User Domain) capabilities support cooperation with other users' avatars. CHIME maps and populates each environment with structure and content drawn from external (potentially legacy) information systems, including SQL/OQL databases, Internet and intranet Web servers, document management systems, and so on. Users invoke their conventional querying, browsing, editing, etc. tools as well as add-on services through virtual world actions in the containing context provided by "rooms", "corridors" or "levels".

CHIME enables organizations and multi-organization enterprises to unify, manage, and quickly utilize disparate, distributed business-critical data and service assets through a highly intuitive group-oriented user interface: no one ever seems to need "training" to socialize in chat rooms or play first person shooter games!



CHIME System Architecture

System Description

CHIME is extremely dynamic, flexible, and fully adaptable to an organization's existing information infrastructure. By providing uniform access via intuitive explorative navigation facilities for group information spaces, CHIME makes it easy for new staff to come up to speed and keep existing team members up to date.

CHIME's ***Data Server*** *(***DS**) uses a JDBC-compliant open source SQL database – HSQL (Hypersonic SQL). DS manages content metadata, not the content itself, so it does not need ownership or control of the data it organizes, hyperlinks, and annotates. An integrated (but separately useful) component, ***FRAX*** (**F**ile **R**ecognize **A**nd **X**MLify), recognizes backend object types and then applies plugs that are individually tuned to extract metadata from that object type and convert it into XML. This metadata is propagated via Siena (from U. Colorado) into the DS where it is parsed and catalogued. FRAX was designed with the goal of being easily expandable, so that developers can add plugs to support other proprietary protocols and data formats.

CHIME's ***Virtual Environment Modeler*** (**VEM**) "tags" DS data elements with an extensible set of Virtual Environment Types. Base types include Component (an atomic data element representing a single piece of information), Container (which may aggregate other "child" data elements), Connector (a logical connection between two or more data elements), and Behavior (human users, software bots, and external tools and services). Administrators define data element subtypes appropriate for their particular environments.

The ***CHIME World Manager*** (**CWM**) generates the virtual world's layout and content from DS data elements augmented with VEM typing. Data elements tagged as 'Containers' may become Rooms in the virtual world, with their child 'Components' visualized as Furnishings inside the Rooms. Connectors might be drawn as Doors, furnished Hallways, or simply popup menus for teleporting among Rooms. CHIME clients render the 3D immersion onto users' displays. Changes to backend data sources are reflected in the virtual world through incremental or checkpoint scene updates, and vice versa (when authorized). CWM also keeps track of the users in particular rooms and object locations in those rooms. Upon any change to objects in any room, updates are broadcast to all affected users.

AI^{2}TV

Introduction

Adaptive Internet Interactive Team Video (**AI^{2}TV**) is an experimental extension of CHIME to support enterprise workflow and synchronized multimedia (particularly video) within the 3D collaborative virtual environment. **PSL**'s ***Kinesthetics eXtreme*** (**KX**) monitoring system (a DASADA project) will be employed to probe video server, proxies and clients, gauge bandwidth, quality of service and degree of synchronization (or lack thereof), and continually coordinate the system components and resources to maintain an effective user experience. **PSL**'s ***Workgroup Cache*** (initially an EDCS project) will utilize a variant of KX's complex event pattern matching rule system to determine when to prefetch (pull) or propagate (push) video segments and other shared information apropos to a collaborating group's task-oriented context.

AI^{2}TV can be applied to a broad range of uses:

> Consider a group of five trainees participating together in a logistics planning exercise. Two of these trainees live on the same base, and access the training organization computers via high-speed Ethernet from their rooms, but the other three are geographically dispersed and use widely varying quality connections from their home or office PCs to "meet" with their team in a virtual world. Their next task is to devise and execute (in simulation) a movement plan in response to a remote "situation". The training instructor has defined a particular structure that this plan must take, and supplied a dynamically changing (canned) database of available resources and their spatial locations, and a collection of (simulated) video feeds from satellite and ground locations, augmented by a training lecture that guides the users through the exercise.
>
> Imagine the trainee meeting starts by reviewing the videotaped lecture in which the instructor explained these materials and appropriate approaches, or just the most germane portions thereof to save time. Other trainee teams have already viewed this lecture as well as the corresponding video feeds, so keywords taken from transcripts of their CHIME chat discussions have been automatically associated with the video segments they were watching at the time. This segment index is depicted as, say, a 3D postmaster's cabinet in a room where the users' avatars meet, and an avatar can select a segment from a pigeonhole to watch on demand. This room also contains 3D depictions of the written specifications for the assignment as well as the trainees' class notes and their in-progress planning documents. High- and low-tech logistics planning tools can be combined in the virtual environment: for instance, interactively constructed partial plans could be projected onto the wall of a room, and/or a "live" flat or contoured map zooming into the area of interest shown on a table - onto which auto-scaled 3D models of logistics assets can be placed and maneuvered.
>
> Since the environment employed workflow agendas and schedules to "know" that the deadline for the assignment is fast approaching, it prefetched the most significant portions of the lecture video onto the trainees' PCs during idle time. This freed the available real-time bandwidth for the simulated video feeds and database queries, whose dynamic results are intentionally different for each team. The lowest-bandwidth (and/or smallest cache) users necessarily see more highly condensed video feed segments – perhaps just a sequence of still shots – but still must maintain semantic synchronization with their higher-bandwidth teammates.

System Description

The AI^{2}TV project has three interrelated experimental components. The first (led by Prof. John Kender at Columbia University) investigates automatic methods for deriving semantic video structure, by finding large-scale temporal and spatial patterns, by detecting redundancies and semantic cross-correlations over long disjoint time intervals, and by compressing, indexing, and highlighting video segments based on semantically tagged visual sequences, each labeled with measures of perceptual and/or cognitive relevance.

The second experimental part (led by Prof. Jason Nieh) designs, and analyzes the experimental impact of a class of particular server cluster configurations, wire protocols, proxies, local client caches, and video management schemes. These accommodate varying latencies, throughputs, client processing power, and server work loads, particularly with respect to the varying semantic qualities of the indexed video data.

The third experimental part (the main focus of **PSL**'s AI^{2}TV work) explores user interaction and information access in Internet-scale distributed environments in which interfaces for the rapid and random access to video is a necessary and dynamic resource, and measures the impact that workflow-based knowledge of present and anticipated user activity, other information - and service-based contextualization, and continual validation (such as KX) can have on multimedia delivery - whether to a shared and coordinating 3D virtual world or to conventional 2D client. These schemes will also manage video segment prefetching and refinement based on anticipated work patterns, past work patterns, and semantic cues present in verbal user interactions via the environment's chat windows as well as other user actions such as database queries and tool invocations.

Features

Unified context for information and services

CHIME virtual environments bring together information from disparate data sources and unify access through the common immersive interface. "Theme"-specific scene graphs and textures may lay out data into rooms and other containing contexts according to missions, business processes, organizational structures, security domains, or any other mapping aimed to assist team members in quickly finding information and tools biased towards the task at hand. Avatars can move holograms of objects from room to room for temporary access/use. PSL's Worklets (see separate flyer) can rapidly and predictably disseminate new processes or services to contexts demarked by individual rooms or rooms groups into "levels".

Geographical/temporal distribution

Physically dispersed team members inhabit a common virtual space, easing the communication difficulties of distributed enterprises. The 3D immersion, and our upcoming 2.5D approximation for mobile handheld devices, enables stronger senses of user co-location, situational awareness, and proximity among logically related materials than older systems like Orbit, eRoom, BSCW, NetMeeting, etc.

Scalability

CHIME worlds may be dynamically connected so that users "walk" between Rooms supporting different organizations or endeavors. From the users' perspective, they inhabit a seamless virtual world, easing multi-team collaborations and information/process sharing. Of course, administrators easily limit access privileges. Users and resources from (logically, not necessarily physically) adjoining spaces might potentially be "recruited" to participate in local activities.

General Specifications

All CHIME server software is 100% Pure Java 2. CHIME servers have been tested on Windows NT/2000, Sun Solaris, and Linux. 128MB RAM recommended.

The CHIME client is written entirely in C++ and utilizes the open source Crystal Space libraries for 3D rendering. The client has been tested on Windows NT/2000, including archaic as well as latest generation video cards. A 2.5D "map" client for PocketPC is nearly complete, and a 3D Linux version is planned.

Availability

http://www.psl.cs.columbia.edu/software/download/