Research Investigating Generation-Beyond-Next Computer Game Culture and Technology:

A Collaborative Research Partnership between the UCI Game Culture and Technology Laboratory and the Daegu Global R&D Collaboration Center

Progress Report

Walt Scacchi, Robert Nideffer, Craig Brown, Yuzo Kanomata, Kari Nies, Alex Szeto and others
Game Culture and Technology Laboratory
Institute for Software Research
California Institute for Telecommunications and Information Technology
University of California, Irvine
Irvine, CA 92697
Wscacchi@uci.edu

Report Period: 1 July 2008 – 30 June 2009

Introduction

The report documents progress and results obtained from our research study that is investigating generation-beyond-next computer game culture and technology during the period of 1 July 2008 through 30 June 2009. This study is in support of a collaborative research partnership between the UCI Game Culture and Technology Laboratory and the Daegu Global R&D Collaboration Center, in Daegu, Korea. The initial scope and research areas for study was agreed to by both partners in December 2006, and that served as the basis for effort initiated during this project reporting period.

Project Plans and Activities for 2008-2009

Our activities in the past 12 months have focused on a number of topics described below. Each has been the subject of previous visits, email discussions, or presentations via teleconference with DIP since July 2008. These include research activities focused on ongoing efforts previously documented in our last progress report to DIP (submitted July 2008): (1) ongoing development and refinement of the WTF?! game engine and software development kit ("!"); (2) ongoing investigations into the development and refinement of methods for integrating media servers with game engines and other back-end game services; (3) use of game modding and other methods for ongoing development and refinement of "serious games" like the *FabLab* game for training semiconductor fabrication technicians and *DinoQuest Online*; (4) ongoing investigation and refinement of open source concepts, tools, and techniques for developing online environments to facilitate collaboration and cooperative work among geographically dispersed teams; and (5) other topics that have emerged from our research effort to date.. Finally, there have also been research activities that focus on the development and refinement of new concepts and approaches for new R&D projects addressing emerging opportunities in computer game culture and technology. Activities in each of these six areas is briefly described below, and followed with a collection of reports that help document what we have learned along the way, and that we can share with confidence.

WTF?!

Our ongoing development and refinement of the WTF?! game engine and software development kit ("!") has been the focus by Robert Nideffer and Alex Szeto. As of July 2008, an initial version of the WTF?! Game, game engine, and game SDK (hereafter, WTF?! 1.0) was publicly released for global distribution, usage, and feedback. As previously noted, the WTF?! 1.0 web site (http://aoedipus.net) had by July 2008 attained a position in the top 250,000 Web sites in the World (out of >>10B sites), as well as news reports and user commentary in more than a dozen languages spanning more than 20 international countries. As user-demand overwhelmed our capability to distribute the game for online play, WTF?! 1.0 was modified and repackaged as a single zip archive file so that it could be downloaded for play by a single user on a PC. This unfortunately meant we could no longer collect online data regarding game play/usage statistics that could help us better understand how online game play was (or was not) proceeding. Nonetheless, Nideffer and Szeto went forward during Summer 2008 to develop specifications for a new version of WTF?! game engine and SDK (but not the game), subsequently called WTF?! 2.0. Much of the effort that followed up to this time and still in progress focuses primarily on the SDK, as many of the improvements in the game engine were first to be realized. This also represents a design choice to freeze development of the game, and to a lesser extent the game engine, in order to focus on a complete overhaul and ultimately new start development of the SDK developed using Adobe Flash ActionScript 3.0 programming/scripting language, which offers a non-upward compatible set of extensions and new implementations of Flash functionality that would best support the WTF?! 2.0. This turned out to be a painful and time-consuming technology migration but the effort invested in now bearing rewards regarding the WTF?! 2.0 SDK. Finally, as WTF?! Was influenced by ongoing play experience with the *World of Warcraft* (WoW) MMORPG, then some effort was also devoted to better understand how user-created mods (or "add-ons") can be produced, as another basis for understanding how to incorporate similar capabilities in the WTF?! 2.0 SDK.

Four reports document effort in this area: (a) one on the emerging SDK now featuring a set of in-game compatible object/content editors; (b) another that highlights the more limited editing capabilities supported in WTF?! 1.0 (see Quest editor, Item Editor, NPC Editor, and Level Editor); and (c) another the documents a set of example WoW add-ons that were created and are available for download and use by regular WoW players. Finally, as another important recognition of the accomplishments of Nideffer and Szeto in developing and deploying WTF?!, the WTF?! 2.0 game has been selected for exhibition at the Laguna Art Museum (http://www.lagunaartmuseum.org) in nearby Laguna Beach, CA as part of the new Summer art exhibit, "WoW: Emergent Media Phenomenom" that feature a selection of artworks by 16 international new media artists, along side of artwork from the development and distribution of WoW by Blizzard Entertainment. Nideffer also created a new artwork called "2007 BC" which is a large-scale digital print which is also part of the exhibition, and is briefly included herein as the fourth report from the WTF?! Game research effort. The last report here (d) documents the 2007 BC artwork. Note the high-rez versions of the 2007BC images are not available online.

Streaming Media Servers for Online Game Environments

Our ongoing investigations into the development and refinement of methods for integrating media servers with game engines and other back-end game services is our next area of effort. During early 2008, we had been focusing attention on how best to integrate a back-end server that could support the capture, archiving, retrieval, and replay of in-game and around-game video streams or videoconferences. Effort has continued by first focusing on how best to evaluate alternative proprietary and open source streaming media servers including the Red5 open source server, which as offered by SilverFox as a commercial-grade MMOG back-end server can be used to support up to 500K concurrent users (vendor claim). Our experience with such servers to date does question what it means to support large numbers of concurrent users along with what streaming services they may actually be using. However, our experience might be limited by our inability to invest large amounts of funding to engage the proprietary server vendors. But it appears that the number of users may be large when the data streaming services are fairly simple (e.g., 2D online game play in online rooms, rather than complex levels or open end virtual worlds in 2D/3D; small format, playback-only video streams that are apparently popular in adult entertainment online services). Nonetheless, understanding the underlying software architectures, capabilities, and current limitations of back-end streaming media servers remains an active area of study by our group. The first report in this section documents our results and knowledge gained to date.

Similarly to evaluating the capabilities and limitations of the streaming media servers and how they may be used to support the capture of live streaming media content (e.g., live game play feeds or live game player videoconferences), we have also been investigating new client-side technologies (and also client-server solutions) that are available in proprietary and open source software formats. A second report in this section documents our results and knowledge gained to date. In addition, we have also investigated these client-side approaches to streaming media capture in the context of online virtual worlds, to help see if there are any other issues that may arise when dealing with multi-user play or collaboration in games or game-based virtual worlds (as there is great commerical interest in virtual worlds targeted to young audiences under 18 years old). A third report in this section documents our results and knowledge gained to date.

Serious Games R&D

Our ongoing development and refinement of "serious games" like the *FabLab* game for training semiconductor fabrication technicians and *DinoQuest Online*. Three activities have been the focus of effort in this area. First, ongoing development and refinement of the *FabLab* game focused on transitioning from development using the Unreal 2 game engine (as used in the *Unreal Tournament 2004* game) to the Unreal 3 game engine (as used in the *Unreal Tournament 2007* game). Once again, the transition involved moving across non-upward compatible game engine and game software development (modding) capabilities and features. Thus, most of the effort to date had to be redone in full or part (e.g., the 3D object models used different modeling constructs, such that all 3D models had to be redeveloped and rescripted). The first report in this section documents our a sample of results to date, such that we have recreated and extended the functionality of the FabLab game from 2008, so that new training applications and features can be supported (not shown).

Next, a new effort was begun to demonstrate the rapid, low-cost development of a serious game for training business analysts who need to learn how to work with complex information systems like Enterprise Resource Planning (ERP) systems, in a large enterprise setting. For shorthand purposes, we refer to this game as, *CBA*, short for computer-based analyst training game. Emphasis here included an end-to-end game development schedule measured in weeks, game programming and content development done by a single person with comparatively little game development experience, interaction with training application domain specialists with little/no prior experience in game development but with experience in business training content development, and use of low-cost (2D) game development tools. As the resulting game is not available for redistribution at this time (due to a confidentiality agreement), we do instead provide a generic version of the storyboard that was developed and evaluate the game and game art assets. Once again, effort here focuses on seeking to learn how little resources and skill can be put into motion to develop and deploy a serious game for corporate training applications. The second report in this section documents this storyboard

Last, we have begun to write-up and summarize some of the things we tried and learned from in the development, deployment, and experience of playing *DinoQuest* and *DinoQuest Online* at the Discovery Science Center. As we have discussed many times before, we believe that "science learning games" are a very compelling and high-value domain for serious games. The third report in this section documents our results and knowledge gained to date that focuses attention on how to design the game and game play experience, whether in a mixed reality environment like the Discovery Science Center, or in an online environment for *DinoQuest Online*, can be designed to facilitate multi-player collaborative play (or work) even among people who have little/no prior contact. The third report in this section documents our results and knowledge gained to date.

Collaboration tools and techniques for dispersed teams of players/developers

Our ongoing investigation and refinement of open source concepts, tools, and techniques for developing online environments to facilitate collaboration and cooperative work among geographically dispersed teams has taken a turn for the better in this area. Our prior effort up through mid 2008 focused on the Virtual Collaboration Portal (VCP) whose development never quite reached a complete 1.0 release. This VCP implementation focused on the integration of eight or so major open source system components, each being developed and released asynchronously to the others. This resulted in a nightmare of software updates and vulnerabilities that could be managed if we had substantial staff resources just to focus on this, but we did not have such staff nor interest. Instead, we revised our effort to instead focus on development using a robust, single open source software content management system, *Drupal*, which is used by thousands of websites worldwide, and has hundreds of active developers who build and integrate different functional modules that extend the core system's capabilities. To date, we have been able to develop or integrate a substantial portion of the functionality we originally sought to incorporate into the original VCP prototype, so that we know have a beta-version of the new VCP up, running, and end-user stress testing with integrated streaming media services now in progress. The first report in this section documents our results and knowledge gained to date.

Next, given our progress with the new version of VCP, we began a new study to explore how such an environment might be integrated with an open-ended game or game-based virtual world. Our reason for this was to explore the hows and whys people might want to bring together online game play/work with streaming media services, so as to be able to capture, store, retrieve, and view/engage streaming media content whether live online or played-back (and annotated) from archived recordings. The next two reports in this section documents our results and knowledge gained to date, with the first focusing on evaluation of alternative open-ended (or open source) virtual world (VW) capabilities, and the second focusing how identifying issues to address when seeking to integrate VCP services within a VW.

Last, another study addresses how we can begin to better envision the capabilities needed to support distributed collaborative work (or play) within a dispersed team of participants. Emphasis here is on teasing out some concerns that distinguish teamwork when players are co-located and coordinated but not necessarily collaborative, and other teamwork/teamplay variations that are possible when new game based approaches, like those from the *FabLab* and *DinoQuest Online* games are considered. A fourth report in this section documents our knowledge gained to date.

Other topics in game culture and technology research and practice at UCI

Next, other topics that have emerged from our project effort in the past 12 months. First, Nideffer and Szeto, along with a visiting game researcher Professor Alf Inge Wang, prepared a report on the architecture and development of the online game-based world of *unexceptional.net*. Though the development of this game world preceeded our project with DIP, this world proposed and demonstrated perhaps the first such online, multi-modal game world that spans a variety of heterogeneous hardware devices and software services including GPS-based cell phones, exiting Websites, 2D and 3D game clients, game-specific blogs and Web portal, and more, all in a manner that allows for continuous game play across these devices and services. The first report in this section documents our results and knowledge gained to date with *unexceptional.net*. This knowledge may become useful in the future should we engage in the research and prototyping of mixed/augmented reality games that involve physically embodied digital game objects/devices (e.g., robotic dogs or robotic plants) who activity/behavior is reflected in an online game-based virtual world.

Second, Nideffer and Szeto in a more recent effort have borrowed concepts and capabilities from the WTF?! 2.0 game engine, along with new features provided by the Adobe Flash ActionScript 3.0 language, to begin experimenting with the development of a new 2D game-based virtual world that utilizes real-time streaming data feeds to drive/influence game play. This in-progress game world is called, *SPEW*. The second report in this section documents our results in progress to date. Once again, experience with this effort may become useful in the future should we engage in the research and prototyping of mixed/augmented reality games that involve physically embodied digital game objects/devices either provide or require real-time streaming data feeds to modulate ongoing game play either in the physical or virtual game world.

Third, Scacchi and other faculty in the School of Information and Computer Science have begun a large, multi-year study of the development and use of virtual worlds, including game-based virtual worlds, within different enterprise settings. Initially five, but now seven enterprises are involved as research partners in this study, including two aerospace/defense firms (The Aerospace Corporation and Northrop-Grumman Corporation), one telecommunications firm (Avaya Laboratories), one transportation start-up venture (UniModal Inc.), one regional science center (Discovery Science Center), one microelectronics manufacturer (Intel), and one commercial VW vendor (EON Reality). Three other enterprises are currently engaged in negotiations to also become partners in this project. Finally, we believe that DIP should become and be recognized as our first international partner in this effort as well, as our current project (and perhaps future projects) with DIP centers on R&D studies centered on multi-player games and game-based virtual worlds, as documented above and below. The research program for this effort is described in the third report for this area, while the fourth report describes how two game research efforts that we have engaged and documented as part of our project with DIP begin to articulate one aspect of the research program.

Finally, we note that the UCI School of Information and Computer Science, along with other faculty affiliated with the UCI Game Culture and Technology Lab, are now moving to form a larger campus research center to focus on culture and technology of Games and Virtual Worlds. More than 20 UCI professors across multiple disciplines have expressed their interest in wanting to be part of this center, as well as to become involved in future research projects addressing various aspects of games and virtual worlds. Our collaboration with DIP and other research partners has helped to advance faculty and student interest, as well as future participation. Also, the growing number of UCI faculty now actively interested in participating in game/virtual world research suggests that larger and more topically diverse projects can be contemplated in the future.

New game concepts and emerging proposal ideas for 2010 and beyond

Last, there have also been research activities that focus on the development and refinement of new concepts and approaches for new R&D projects addressing emerging opportunities in computer game culture and technology. As this effort represents the beginning of one or more possible future research projects, the one report in this section outlines a set of possible projects for which we, UCI and DIP, can collaborate on researching starting in the last part of 2009 or soon thereafter. As the material in the report for this section have recently been discussed at length via videoconference and associated email correspondence, this material is included here simply to document where we started from and perhaps where we may be going. However, ongoing discussion and collaboration will determine which topics will emerge as the basis for our next project(s) between DIP and UCI.

Final Remarks

With this preceding project areas in mind, we now turn to present the materials that are included in the remainder of this progress report, which help to document what we have accomplished and learned during our effort from 1 July 2008 through 30 June 2009.

The remainder of this report is far more voluminous than this overview. This is as it should be in our view. Whether the diversity and volume of results that follows is indicative of what will be produced in the research period ahead is unclear. So please view the remaining materials with an eye towards what research or directions might be investigated in the months or years ahead. We will of course have the opportunity to discuss and review these materials through video-conferences and other meetings during the remainder of 2009.

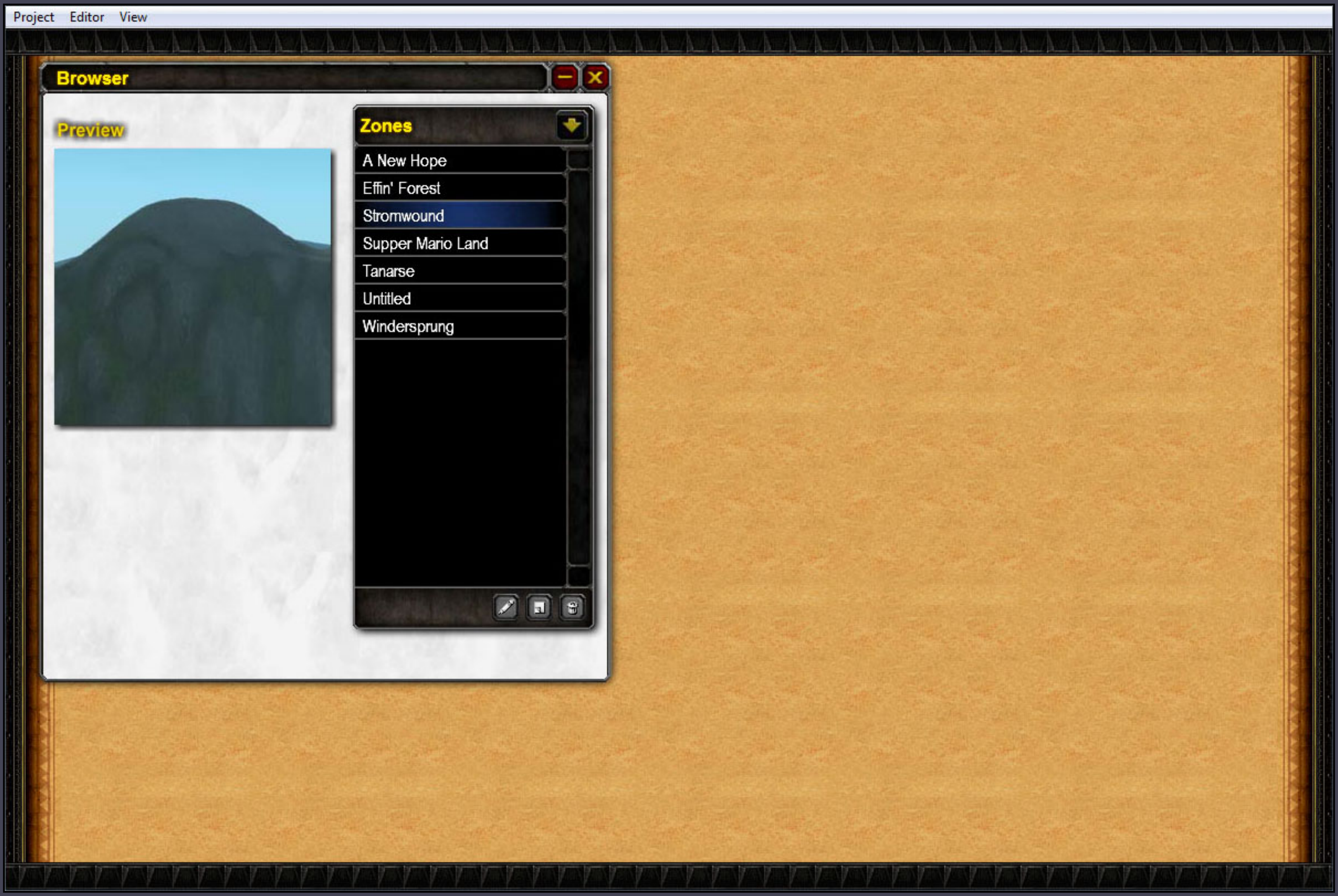
WTF?!

"!" - Game Modding and Development Kit

(A Work In Progress)

'08-'09

Asset Browser



- WoW inspired side-scrolling action RPG game modding and development environment
- Built in Flash using Adobe Air
- Opens to a base project canvas that manages multiple game editing windows
- Asset Browser previews all game assets and launches editors

Level Editor

- Photoshop style editing mode
- Import any number of layers to place objects
- Adjust scroll speed of layers independently for parallax scrolling
- Create custom weather conditions with a percent chance of occurring
- Click and drag to scroll zone and place objects, characters, spawn points, spawn conditions, etc
- Minimap supports placement and manipulation of objects as well

Color Picker

- Assign each layer unique graphics and color tinting
- Assign key-framed time of day tinting changes based on a 24 hour clock that automatically interpolates color values

Quest Editor

- **Supports creation of multiple quest types - collect, kill, escort, object initiated, chained, etc**
- **Establish requirements for access, rewards, items needed for completion, and more**
- **Define relationships to other quests**
- **Create quest text for introduction, objectives, progress, completion**

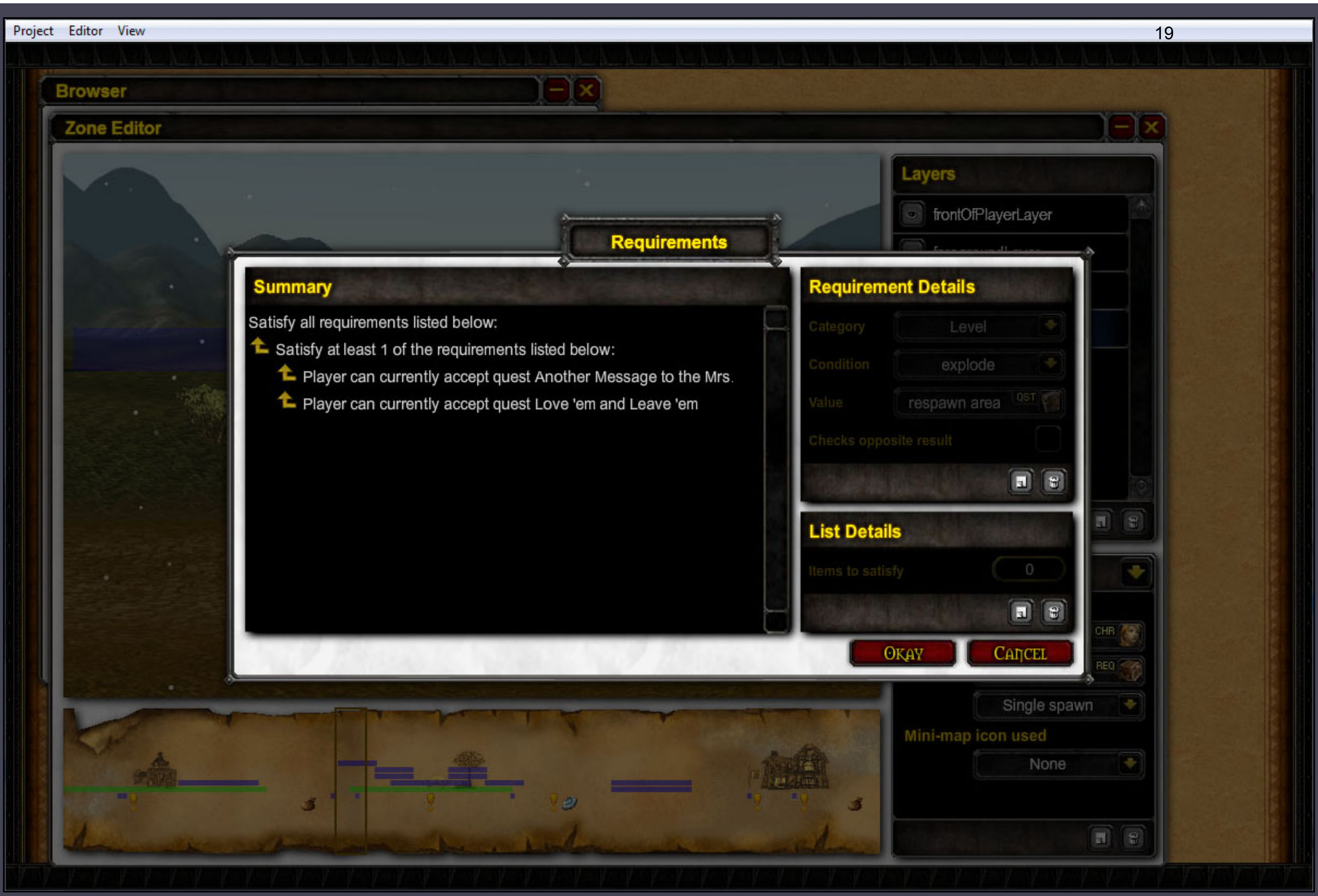
Item Editor

- Create items associated with quests and random drops
- Set descriptive "flavor text"
- Assign quality, type, subtype, level requirements, cost, bonuses, ability cost / effects

Ability Setting

- Set item abilities such as type, whether item effects stack, stack limits, duration
- Link associated graphics
- Create descriptive text summarizing abilities for player

Requirements Setting



- **Establish hierarchies of requirements that must be met prior to items and quests becoming available**

Character Editor

- Create custom characters
- Choose attributes and behaviors based on pre-defined templates or create your own
- Establish identity markers (faction, profession, race, class, and anything else you want)
- Attach pre-set animations or create your own
- Create character tinting effects based on state
- Determine default equipped gear
- Author chat dialogs
- Define complex sets of behaviors
- Set loot tables
- Decide if character is a quest giver, and which quests are available if so
- Preview the settings

Projectile Editor

- Choose from a seed particle list to build more complex projectile effects
- Set properties for projectiles (name, duration, damage, trail effects, movement patterns, etc)
- Preview the settings

Particle Effects Editor

- Choose from a seed particle list to build more complex particle effects
- Associate these effects to projectiles or other items and conditions
- Set properties for particles (whether static or dynamic, behavior, appearance, amount, duration, etc)
- Link custom made sounds to particle effects
- Preview the settings

Animation Editor

- Browse and link a series of image frames to custom character animations
- Associate animations with behavior states (idle, jump, attack, stun, death, cast, etc)
- Define which frames get used
- Set bounding boxes to establish impact areas
- Preview the settings

Capability Editor

- Link animations defined as actions to character states
- Adjust the details of those states (movement speed and angle, associated sounds and particle effects, damage conditions)
- Preview the settings

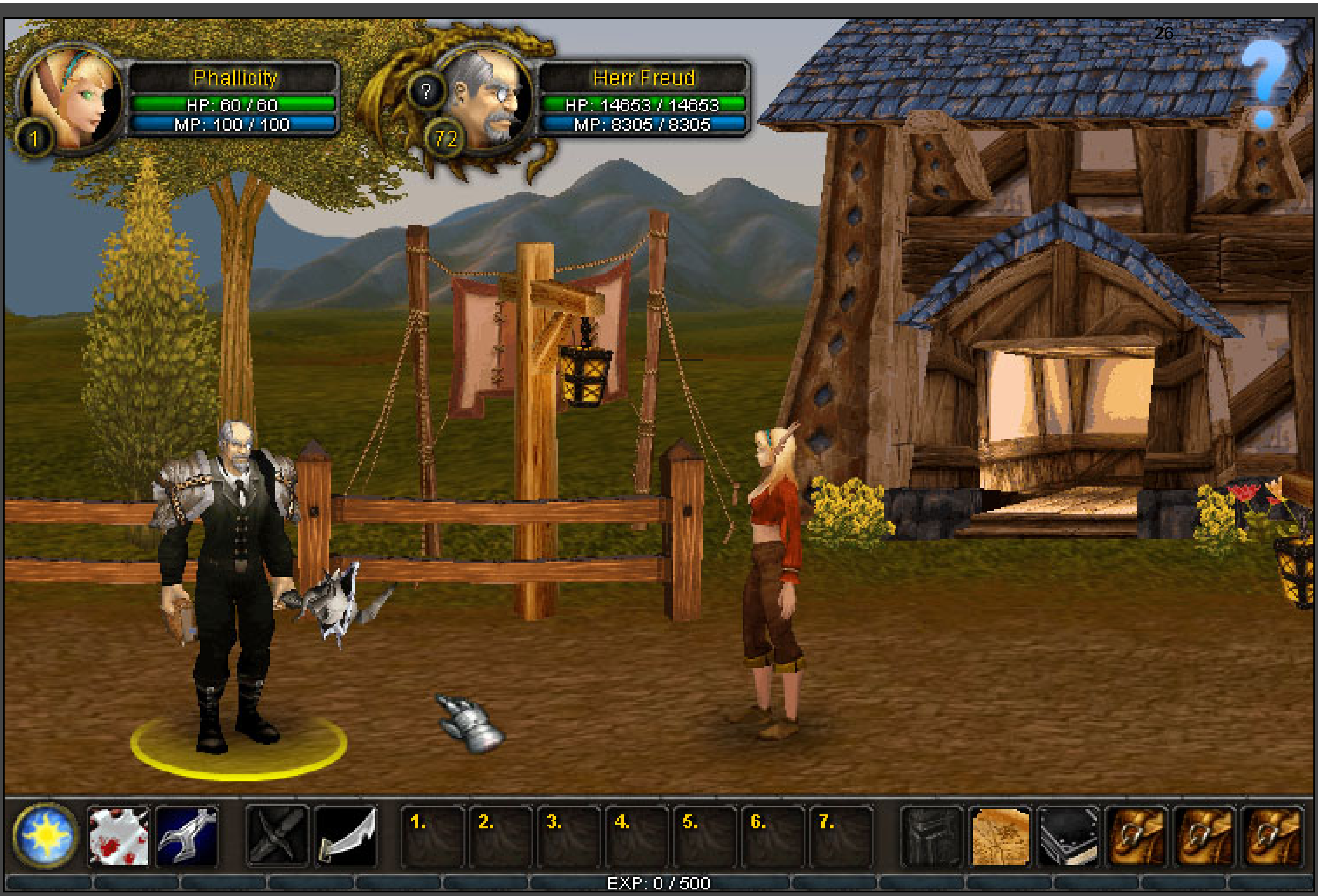
PORTAL

WTF?!
'07-'08

Character Selection

- Side-scrolling action adventure RPG, "World of Warcraft" knock-off
- Do "game theory" in the context of a game environment
- Select from 2 fully playable characters in the demo version
- Meet famous thinkers like Karl Marx, Sigmund Freud, and Mary Daly
- Quest in 4 unique zones
- Interact with over 15 NPCs
- Advance to level 10
- Download a freely available toolkit to mod or build your own RPG

Herr Freud



- **Support for terrain maps with parallax scrolling**
- **Customizable NPC behaviors**

Hegemon - Quest Giver

- **Quest system supporting kill, collect, escort, and special types**
- **Automatic objective tracking**

Mr. Marx - Character Profile

- **Slot based item drops that evaluate drop list of each slot independently**
- **XML script support for custom icons**
- **Equipment slot and rarity (common, uncommon, rare, epic) definition**

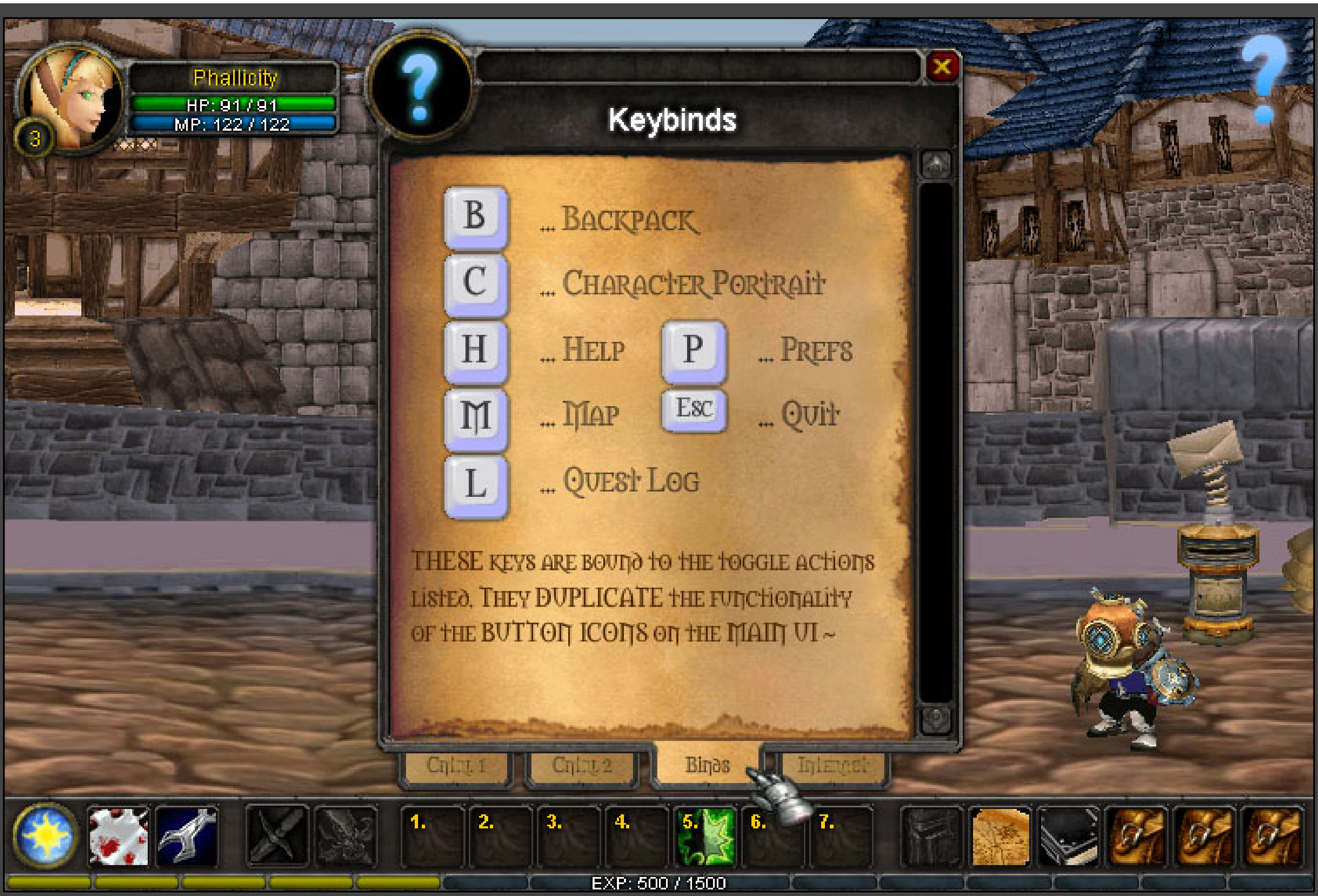
Anna O. - Vendor

- **Ability to define any NPC as an Item Vendor**
- **General equipment slots (head, neck, shoulders, fingers, etc)**
- **Trinkets (activated by casting, reusable, interruptible)**
- **Consumables (activates abilities instantly and consumes item)**
- **Usable Quest Items (activates instantly, does not consume item)**

Game Controls

- Sophisticated fast-paced combat system
- Complex inventory and equipment management
- Ability to follow, evade, attack, throw, change equipment, and activate trinkets

Keybinds



- Shortcut keys to access all important game dialogs
- Shortcut and hot-key access to inventory / equipment items

Controls for Music, Weather, Particle Effects

- **XML script support for all environmental definitions**
- **Particle effects system**
- **Time of day definitions with customizeable tinting**
- **Color tint blending based on the player's system clock time**
- **Weather table and weather specific layer tinting**
- **Background music definition and auto fading during map transitions**
- **Event sound definition attacched to NPCs, players, and items**

Interactive Context Map

- Basic awareness of distance between characters and player
- Objects can be set as key points for display on in-game map

Spell Casting

- **XML script support for pre-built static animations**
- **Particle spawning and respawning**

Portal Transport

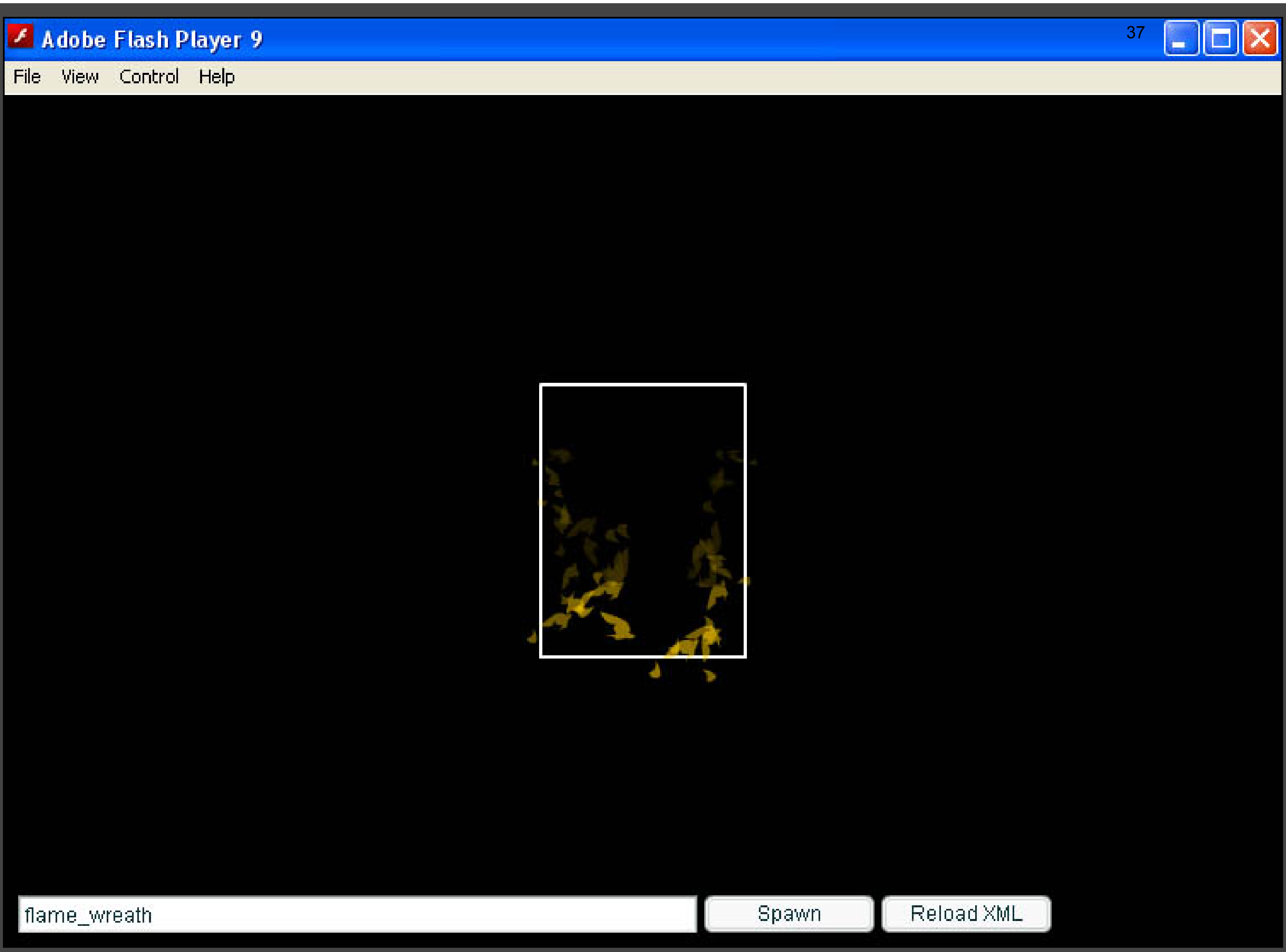
- **Ability to define any NPC as a Portal Master**
- **Ability to instantly transport between zones once visited**

Mario Land

- **And much much more...**

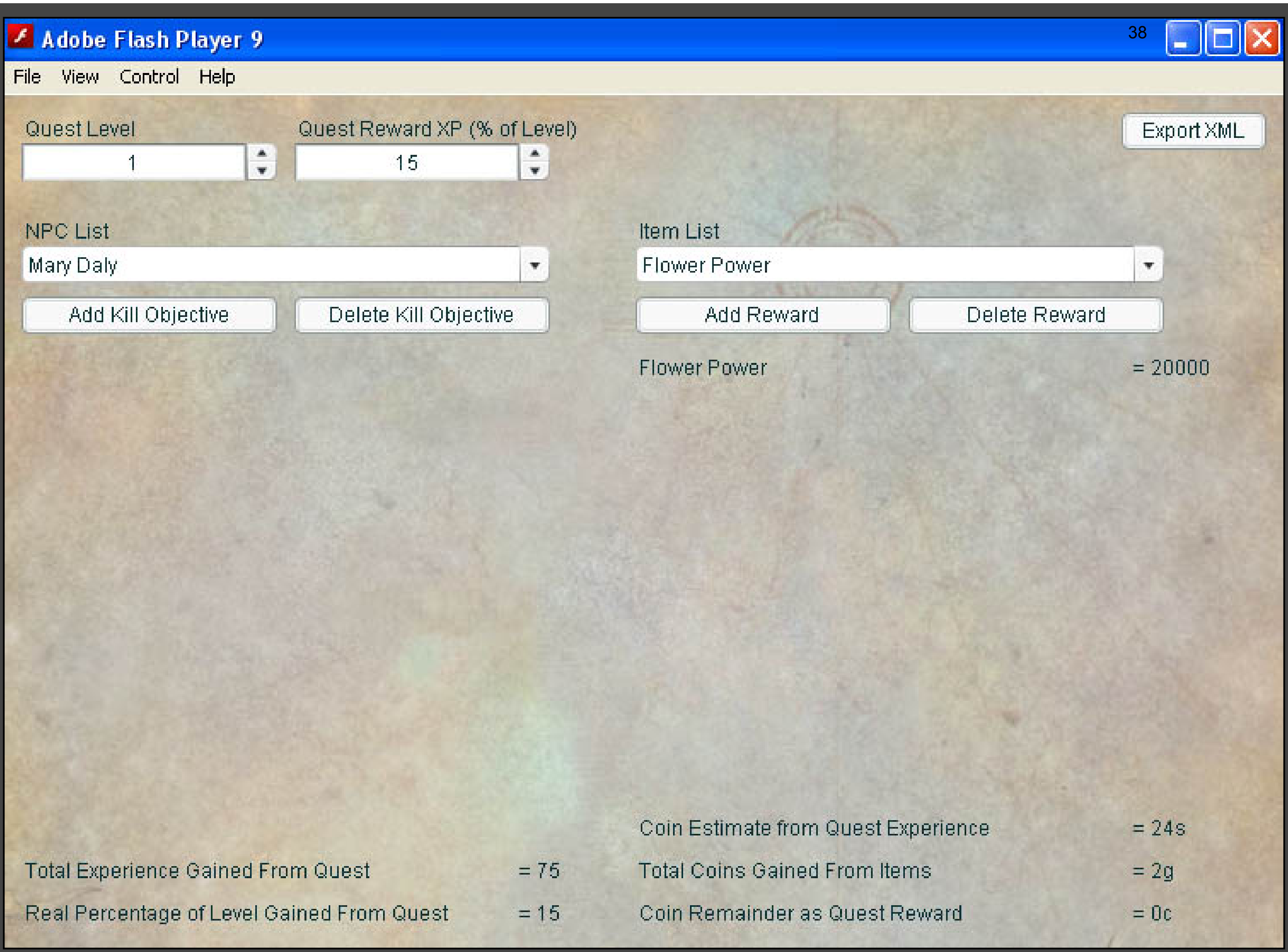
Developer Tools Examples

Particle Effects Editor



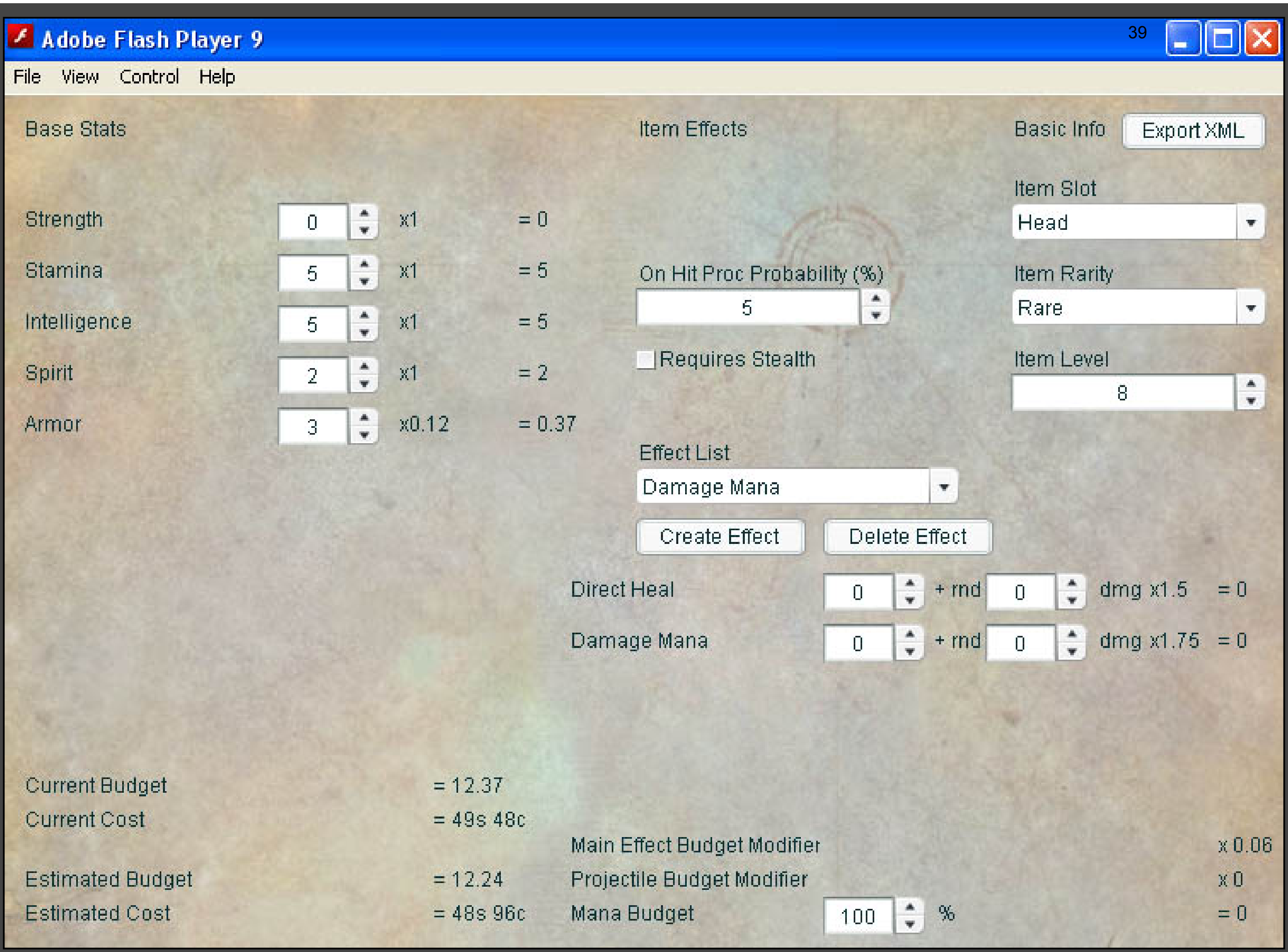
- Design and test custom particle effects
- Highly optimized - based on duplication of single seed image
- Trailing effects, impact effects, splash effects
- Damage range, impact force and angle
- Ability to attach sound to particle effects
- Reaction to hitting ground (disappear, stay, bounce, splash)
- Definition for particle trajectory, acceleration, initial size / alpha / angle, and change over time

Quest Editor



- **Completion rewards (gold, experience, and items)**
- **Ability to do identity, level range, and progress checking**
- **Support for grouped check lists**
- **Automatically export quest as XML**

Item Editor

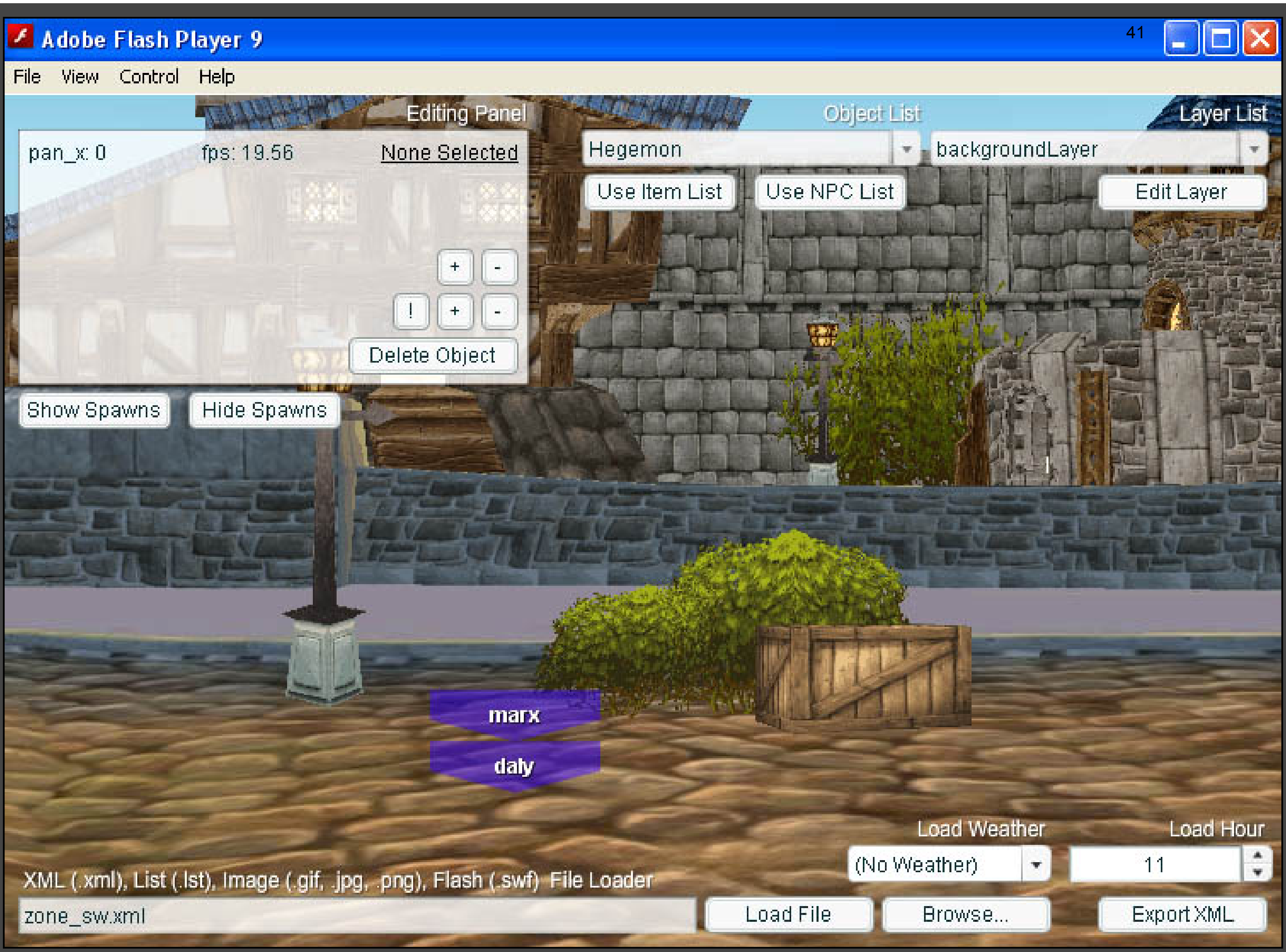


- **Define Item stats for any equippable slot**
- **Projectile Firing**
- **Direct damage / healing**
- **Damage / Healing over time**
- **Damage absorbing shields**
- **Temporary effects on basic character statistics**
- **Temporary / Permanent summoning of other characters**
- **Temporary run speed / jump height adjustment**
- **Control effects (fear, stun, incapacitate, banish)**
- **Stealth Activation**
- **Automatically export item as XML**

NPC Editor

- **Test items in "Paper Doll" editor to do NPC item / level balancing**
- **See how item stats change based on class type**
- **Automatically export NPC as XML**

Level Editor



- Define any number of looping background layers per zone
- Define independent speed of layer scrolling
- Import and click-drag objects to position in real-time
- Define spawn locations for items and characters
- Specify exit locations for transition to other zones
- Randomize NPC level range and density at spawn point
- Set weather table and layer specific special effects
- Automatically export level as XML
- And much much more...

PORTAL - GO LIVE

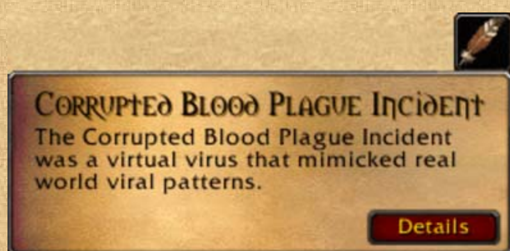
elcome! You have found the **aoedipus.net** distribution and download site for **WoW AddOns**. Check back periodically for **updates**!

TravelGuide

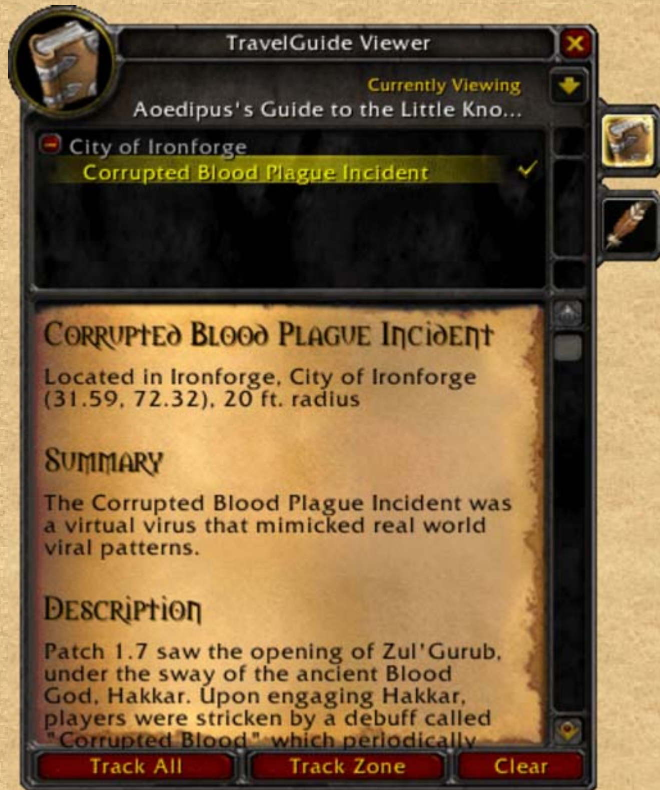
The TravelGuide AddOn lets you Learn about **key points of 'interest'** in custom guides linked to the in-game map. **Create your own guides** to share with others!

TravelGuide Creator: Create your own custom guides

TravelGuide Entry Editor: Create guide entries



TravelGuide PopUp: Displays summary info upon destination arrival



TravelGuide Details: The full guide entry

Information - Installing and using the **TravelGuide AddOn** for WoW.

Install by dropping the "TravelGuide" folder and its contents into your "World of Warcraft\Interface\AddOns\" folder. The AddOn will now be available the next time you start WoW. Simple as that.

You can access TravelGuide through our custom **minimap icon**, or by using the following **In-Game Commands:**

Type: "/tg" or "/tguide" or "/travelguide" to load the guide UI

Download - The latest version of the **TravelGuide AddOn**.

Click the button below to get the most recent **TravelGuide AddOn** for WoW (~55k .zip file). Compatible with all WoW 3.0+ patch content. Also available for download from **curse.com**.

SoundMashup

The SoundMashup AddOn lets you **Map WoW sounds to keyboard keys** to **play and/or share** your own musical mashups with others!

SoundMashup Deck Editor: **Create your own custom sound decks**

SoundMashup Deck Player: **Play with yourself :) and/or with others**

Information - Installing and using the **SoundMashup AddOn** for WoW.

Install by dropping the "SoundMashup" folder and its contents into your "World of Warcraft\Interface\AddOns\" folder. The AddOn will now be available the next time you start WoW. Simple as that.

You can access SoundMashup through our custom **minimap icon**, or by using the following **In-Game Commands:**

Type: "/sm" or "/mashup" or "/smash" or "/soundmashup" to load the editor

Download - The latest version of the **SoundMashup AddOn**.

Click the button below to get the most recent **SoundMashup AddOn** for WoW (~90k .zip file). Compatible with all WoW 3.0+ patch content. Also available for download from curse.com.

xTalk

The xTalk AddOn lets you use a **secret language** to chat to members of the **opposing faction** who also have the addOn installed. It's unclear exactly where this addOn originated from, since **authorship is not claimed**, but it's pretty cool so we thought it worth sharing.

Information - Installing and using the **xTalk AddOn**.

Install by dropping the "xTalk" folder and its contents into your "World of Warcraft\Interface\AddOns\" folder. The AddOn will now be available the next time you start WoW. Simple as that.

You can access xTalk by using the following **In-Game Commands:**

Type: "/xt on" or "/xt off"

Download - The latest version of the **xTalk AddOn** for WoW.

Click the button below to get the most recent **xTalk AddOn** for WoW (~3k .zip file). Compatible with all WoW 3.0+ patch content. Also available for download from **curse.com**

QQ

The QQ AddOn lets you **spout pirated quotes from famous people** who like to complain... **yarrr!** Take your in-game **World of Warcraft** chats to the next level. Look smart and impress those around you with your worldly wit and wisdom! Or just irritate the hell out of 'em for the fun of it.

QQ: **A typical randomly generated comment/complaint**

Information - Installing and using the **QQ AddOn**.

The **QQ AddOn** is a composite of several existing AddOns - the Chuck Norris Fact Generator (by Edgar De Loa, and as modified in Konfuzius by Martika), PirateSpeak (by Blaquen), and AutoLog (by Eternally777). By default pirate talk is on and chats are auto-logged for future reading enjoyment and sharing. However both may be toggled on/off.

Install by dropping the "QQ" folder and its contents into your "World of Warcraft\Interface\AddOns\" folder. The AddOn will now be available the next time you start WoW. Simple as that.

In-Game Commands:

QQ Module...
Type: "/qq say" or "/qq s" (yell, party, guild, raid, officer, raidwarning also can be used)
Type: "/qq whisper [player]"
Type: "/qq [channel number]" (1, 2, 3, etc)

Pirate Speak Module...
Type: "/ps [on/off]" to toggle pirate speak module (on by default)

AutoLog Module...
Type: "/al chat" to toggle auto-logging of chat on|off (on by default)

Download - The latest version of the **QQ AddOn** for WoW.

Click the button below to get the most recent **QQ AddOn** for WoW (~13k .zip file). Compatible with all WoW 3.0+ patch content. Also available for download from **curse.com**.

Download

Note: None of these addOns are officially sponsored by or directly associated with either the **World of Warcraft** developers at **Blizzard Entertainment** or their publisher and parent company **Vivendi Universal**.

2007 BC

'09

Exterior Panels

[Hi-Rez Version]

1. Digital painting produced as a large format (9' x 4.5') triptych
2. Giclee process printed directly to prepped 6mm Baltic Birch panels
3. Custom built hinged Poplar frame treated to look aged
4. Inspired by Bosch's [Garden of Earthly Delights] and [The Temptation of St. Anthony]
5. The exterior depicts world creation

Interior Panels

[Hi-Rez Version]

1. Depicts epic struggle between the world's top two raiding guilds (Death and Taxes and Nihilum) after release of [The Burning Crusade] expansion pack to [World of Warcraft] in 2007
2. Left Panel: World first boss kills by Death and Taxes
3. Center Panel: Intermingling of core guild members
4. Right Panel: World first boss kills by Nihilum

Center Panel Detail

[Hi-Rez Version](#)

1. **References in-game and around-game guild drama**
2. **References themes appearing in Bosch's early work through the lens of an immensely popular contemporary game and social networking platform**

[PORTAL](#)

Streaming Media Servers for Online Game Environments

Reconciling Streaming Server Architectures for Casual Multiplayer Games

Yuzo Kanomata and Walt Scacchi
Game Culture and Technology Laboratory
Institute for Software Research
University of California, Irvine
+1-949-824-5896, +1-949-824-4130
ykanomata@gmail.com, wscacchi@ics.uci.edu

ABSTRACT
Multiplayer games constitute a large and interesting area of both research and commercial interest. During the development of casual multiplayer games, there is often need for static content, and also streaming services. This need included both provision of around game documentation and communication services, in game event data streaming, as well as in game video clips and services providing support for information about the game. In this paper, we describe our experiences and lessons learned in identifying and reconciling architectural mismatches that can arise when integrating heterogeneous game services and clients that combine various types of game play events and experiential media.

Categories and Subject Descriptors
K.3.2. [Computer science education], K.8.0 [Games], and D.2.11 [Software Architecture]

General Terms

Documentation, Performance, Design, Experimentation.

Keywords
Game development, software architecture, game servers, streaming servers, game infrastructure

1. INTRODUCTION

Our group at the UCI Game Culture and Technology Laboratory has been involved in research and development of casual multiplayer games for the past five years. During this time, our group has produced and released a number of such games, including *unexceptional.net*—a pervasive multi-modal, multi-media game [1], *Dino Quest Online*—a science learning game environment for informal science education [2], *FabLab*—a game-based training simulator for semiconductor fabrication [3], and others. Over time, we have found recurring need to produce, publish, and share these games with online Web documents, video conferences and recordings, text/audio chat transcripts, and more to help support game play. In order to provide these services, we used a commercial media server system. However, we sought a lower cost server alternative for both internal and external developers, end-users, and game content creators.

Starting in late 2007, a new game, *WTF?!* was designed as a 2D, side-scrolling game parody of the *World of Warcraft* (WoW) game by our colleagues Nideffer and Szeto, as a next step in advancing our game R&D [cf. 1,2,3]. Of course, to call WoW a game is an understatement, as it embodies a global culture of millions of game players and thousands of Web sites where WoW players, enthusiasts, and fans create and share an endless collection of online narratives and related media, all in tune with their experience and desires. Thus, the design of WTF?! was envisioned to engage would-be players in game play and online game play, communication, and related user created content sharing experiences. The WTF?! game was designed to be an open source Flash-based game engine that allows for developers to create their own side-scrolling games, or modify/remix the content of existing ones. A streaming server system architecture was sought to provide services to help developers. These sought after services, which we referred to as around-game services, included game-based video tutorials, text and video chat with developers, capture of user generated screen shots, or replayable recordings of game play session. Additionally, a multiplayer component to the game was envisioned in our long term development plan. We had previously developed a game portal to provide many such around game services separate from the game server itself. This portal took a federated approach (it used multiple individual servers to provide each kind of service) to provide services. While successful, each individual server required extensive on-going maintenance.

Opening screenshot from WTF?! [3].

In an attempt to satisfy both game needs and to simplify the maintenance requirements we sought to construct services around a server which could satisfy the needs for both in game and around game streaming services. An examination of servers was conducted to see if a suitable means of providing streaming services could be found. We centered our development around Red5, an open source Flash Media Server alternative. But this choice was found to be problematic through architectural mismatches and misalignments, which are the core technical focus of this paper. These problems arose from the heterogeneous architectures that we were trying to integrate as well as the diverse in-game and around-game services we wanted to support. Thus, the examination of these problems and how we sought to reconcile them are the technical contribution made in this paper.

With this in mind, we turn to review prior efforts that influenced our approach, the architectural mismatch problems found along the way, and the proposed solution we have prototyped to demonstrate a way forward with a homogeneous server system architecture.

2. RELATED RESEARCH

Our efforts draw on the different lines of related game server research: game server architectures, in-game and around-game services, and software architecture principles.

First, in developing any multiplayer game, there is need to choose a viable, and ideally scalable architecture that meets the needs and available resources of the game development team. Three broad architectural patterns are common, including:

> *Unified single server* – where a single monolithic, or preferably modular and extensible, server architecture provides all the services required, including game data event streaming from "fat" game play clients through a "thin" networked game update server.
>
> *Peer-to-peer* (p2p) – a game server architecture where each client engine can negotiate and interoperate with each other. This can reduce central server loads, but can be limited by the requirements of the weakest interconnected system.

> *Multi-server/Federated system* – a game server architecture in which aspects of the game are distributed across a set of servers that can collectively interoperate to provide the desired services. An architecture where each server controls a region of game play (or "shard") is an example. The complication in such an architecture can come from coordinating game player events across region transition boundaries, coordinating character transitions between servers, and balancing load across a grid of game compute servers/blades, just to name a few [5,6,7].

A lot of attention is being devoted to such multi-server architectures, as the needs of large commercial massively multiplayer online games (MMOGs) appear to require such processing power to handle large numbers of concurrent game play event streams (one per player, an usually dozens of concurrent players per server or game level). However, for a small, start-up game development studio or a game research team, such processing power may not be available.

Consequently, we face the problem of how can small game development teams be accommodated without major resource investments, and what do they need to know about what's involved in designing and implementing a modern multiplayer game server of any kind that best meets their needs?

Second, the collective notion of what is a computer game seems to be expanding. For example, virtual worlds like *Second Life* to many people are an embodiment of game culture, game level look and feel (e.g., textured 3D geometric objects and avatars), game play mechanics, and playful game experiences. Second Life relies on user-created or user-modified content to form its virtual worlds for exploration and interaction. It relies on a streaming server architecture to continually update end-user views into a shared world (or island), and this in turn facilitates the integration of secondary streaming media services or applications (e.g., in-game video players) to deliver dynamic content accessible to players in-world [8]. The overall scalability of their current architecture has been questioned, in part due to a confluence of business and technological interests coming together (or pulling things apart). Subsequently, a new research community has emerged around the *OpenSim* project [9] that seeks to both explore virtual world (or "metaverse") concepts and techniques, as well as provide open source software architectures to help open up why and how its software servers and clients operate the way they do.

Elsewhere, multiplayer online games, especially MMOGs, are increasingly seen as a new social media for culture formation, performance, and enactment [10]. Players increasingly come to expect that part of the reason they want to play in multiplayer online games is to learn about the how and why to play together, and to learn new things about emerging virtual worlds, often with others they have never met [11]. Consequently, what we call the around-game experience of online socialization and communications (communication outside of game play like posting to a game wiki or player group blog or online discussion forum that is substantively related to in-game play or related situations) is becoming as central of a service feature of MMOGs that no one would now consider deploying a new MMOG without a Web-based game communication and socialization portal. Similarly, the cultural experience for more avid enthusiast game players now includes creation of game-based media like music videos and machinima movies that remix or mashup game play video sessions that have previously been recorded [12]. To us, this begins to signal that the around-game experience has the potential to become an additional new medium, much like the recent emergence of "social TV" that blends broadcast/podcast TV content distributed over the Internet are being multiplexed with other online communication services [13,14]. As such, our understanding of what needs are to be addressed in a future-oriented multiplayer game server must be framed to support both in-game and around-game services with different electronic media.

Last, the world of software engineering and software architecture does help shed some light on what kinds of issues we may now face going forward in designing new multi-media game servers. For example, it is likely that simply trying to integrate different types of game-based event data, Web content, and streaming media servers is likely to lead to a heterogeneous system architecture where many system components, data connectors (e.g., middleware services, data communication protocols), and object types will mismatch or misalign with one another [15], independent of what type of game server architecture is assumed as a starting point. For large game development studios, such an undertaking may just be the cost of doing business, as well as a erected barrier to entry to competitors. However, other architectural principles like that of the modern Web with its REST architectural patterns [16] and potential successor with Computational REST (CREST) [17], point to the potential for the development of coherent, tractable, and homogeneous architectural solutions that support dynamic and reconfigurable media stream mashups may be possible in the future (but not at present).

Thus, our effort in this study is to examine the issues and mismatches that arise in the development of an open source multiplayer game server that builds from known MMOG architectures, accommodates both in-game and around-game services, and seeks to adhere to modern architectural patterns that can give rise to a open source, homogeneous server architecture. We further direct our effort to casual multiplayer games that can support 2D (Flash-based) games, as a way to lower the floor of our entry into the problem space, as there are a number of casual game-based virtual worlds in operation (e.g., *Habbo Hotel, WebKinz* and *Club Penguin*) and in development that seek support for millions of players playing and engaging in various in-game and around-game activities.

3. APPROACH

The design of game infrastructures (e.g., servers, shared data representation formats, communication protocols) concerns aspects of performance and scale. [24] Streaming is an aspect by which game data (state) can be distributed as well as game artifacts (video, audio, etc). In-game services are often architected for game infrastructures to maximize performance aspects of the system (data streaming) while supporting around game services (media streaming) are often excluded. This leads to hybrid architectures where the game servers support in-game action and a new separate host of servers are used to support additional around-game services such as game replay video and developer interaction.

Initially, we began to survey systems to get an indication of how to construct our services and to examine which architecture would be appropriate for our needs. This concentrated on systems that provided a unified single architecture for Flash-based games. During this examination, we found the open source *Red5* media server [http://osflash.org/red5] that serves as a component of the *SmartFox* server (the game server supporting *Club Penguin*) emerged as a candidate which supported the services needs for the current game development in a unified single server.

We had previously developed a game portal (the portal that runs the UCI GameLab at http://proxy.arts.uci.edu/gamelab/portal/) based on a federated architecture that included a Flash Media Server as a component in the system. This portal collected what was then state of the practice tools integrated with a thin php service/unification layer to support around game services.

3.1 Examination of servers

Initial developments of the WTF?! game indicated that a server providing streaming services would be beneficial if the game was to transform from the Version 1 single user game [3] to a Version 2 multiplayer game. An examination of Flash-based games led to some candidate servers. These included the commercial servers *ElectroServer*, *SmartFox*, and others that are being used in large-scale game-based deployments. These servers provide infrastructure on which game environments such as *Webkinz* and *Club Penguin* have a proven deployment history. Embedded in the core of *SmartFox* is a distribution of *Red5*, an open source media server which replicates functionality found in Adobe's *Flash Media Server* (FMS).

FMS provides services for supporting streaming of media (video and audio through the RTMP protocol), but comes with a large per unit cost (licensed by number of user clients). *Red5* provides similar services but is open source (GPL) that in the new release address cluster-ability to allow for potentially scaleable operations.

For the WTF?! game's initial release as a single player Flash-based game, the streaming needs were restricted to providing supporting media to assist interested parties in further developments of the game and for experimentation with around-game service development. Red5 provided the simplest path to incorporate streaming in the context of utilizing it in a Federated Architecture similar the game portal – providing FMS-like Flash-based media services (using RTMP for streaming audio and video in Flash) at low cost as well as potentially providing services for transferring the single player game executable itself (using the built in HTTP services).

A two pronged strategy emerged as a result of the study: using Red 5 as a replacement component in a Federated system such as our previously developed portal where around game services predominated; or, using Red5 as a media server and http server supporting the game as a single server vehicle.
The two pronged approach strategy reflected the two different ways of architecting a game supporting server. The goal for adoption in an existing federated architecture is to conform the Red5 server to be as interchangeable as possible with the existing infrastructure. This implied that services be constructed as a light weight or drop-in structure. The second goal implied the creation of services within the Red5 server infrastructure and build up services. This corresponded to the goals of a unified architecture. While WTF?! remained as a single player game, the first approach was the most desirable, while the second approach was desirable for the long multiplayer game if game data exchange could be developed as well.

3.1.1 Details of Initial Server Load Examination
Red5 is a Java-based system that provides similar functionality as Flash Media Server. Because the game (WTF?!) itself was in programmed in Flash, the around-game media streaming services were constructed in Flash. Red5 provides services through two mechanisms. The first is through a Java-based middleware interface to flat files and applications. This middleware is incorporated through Jetty or Tomcat, which are known as Java servlet container applications. Each servlet is considered a server application by Red5 that can be accessed by a connected client. These containers provide the front-end through which simple HTTP services can be accessed. The second mechanism is providing an RTMP server that replicates the streaming functionality provided by FMS. An initial set of load checks on video streaming (through the RTMP service) was done to test scalability aspects of the server.

To facilitate this study, *Opsview* was selected for performance modeling of the server. *Opsview* is a freely available system used by system administrators typically to monitor large sets of servers in server farms, but Opsview also can be used to study individual server loads and system performance.
Results showed low resource consumption by the server for video loads, but high CPU use in the

client due the Flash Player. For instance a modern dual-core 2.4 GHz CPU running Flash Player in a browser would consume 45% of client CPU usage on simple plays of videos. No such corresponding jumps occurred on the server, just small 2-5% jumps in CPU and small increases of memory usage. So there was a significant start-up cost for the first thread, but a much smaller incremental performance increase as more client session (threads) were added.

3.2 Development of services

The low server CPU and memory loads in the initial investigation indicated that Red5 could provide the streaming services sought. The development effort went forward on a service for video play-back, video recording, and video chat/conferencing. This pursuit was designed as a test of concept in Red5 with most of the effort devoted initially aligned to be a drop-in replacement of services in our pervious portal. During development, some problems arose.

Issues here included what kinds of services would be the best to develop and deploy. Outside developers have developed services for a Red5 environment, but were tied into infrastructure that differed from our game portal. Rework on core portal components such as authentication and user tracking was undesirable as they affected other components of the system. Another issue was that desirable technologies that were in the roadmap for Flash dictated that we pursue newer technologies based on Actioscript 3 (AS3). This meant that little of our preexisting Actionscript 2 based client code nor many open source projects could be used as is directly.

3.2.1 Server-side application development

Red5 provides a framework in which applications are built. This framework is written in Java and is hooked to a web front-end through the use of both Jetty and Tomcat servlet container applications (which act as middleware). Development of server applications must use this framework to build applications connected to Red5. This differs from the application service creation mechanism in FMS which uses a different scripting language. Direct translation of applications from Red5 to FMS or vise versa requires a program translation effort.

3.2.2 Problems in understanding Red5

Red5 consists of a Java-based core on which the streaming server's functionality is based. Some aspects of the server interaction are embedded into the structure of the server, so some interactions occur in particular orders.

The ordering of how Red5 applications attach, start and stop are well documented, but interactions with regard to streaming are not. Unless one is familiar with expected patterns of behavior for FMS, the interaction pattern of Red5 are not obvious.

1. The server expects the semantics of the streaming class interactions to be correctly implemented by the client. The client is expected to have embedded in itself callback functions to deal with standard calls from the server. One such call for stream is `onBWDone`. The purpose of this function is not well understood or explained on the Red5 email lists. The most common work around is to have a null op function. `setID` is another automatic callback where client response is expected.
2. Stream events do not always get broadcast in the pattern one may expect. E.g., when an application on the server serves two clients, the disconnection event of one client is not ordinarily transmitted to the other client.
3. The semantics of sharing remotely (via remote shared objects) are not well documented. Errors made on remote objects are particularly hard to debug as they run across machine and process boundaries and are thus not amenable to normal debuggers.

4. While documentation exists for Red5, information about interacting with streams is particularly lacking. One of the best sources for understanding Red5 is the FCS/FMS migration guide [25], yet this document is silent on how streams work. The server embeds the stream “publish” routines in itself. Therefore, unlike explicit invokes to the server, these functions are part of the server’s functionality. Yet since there is little to no documentation as to what the interaction pattern is, a novice must look toward patterns from FMS examples. The Red5 API for example does not specify what the modes of the stream publish method are, so operations such as “record” or “live” are referred to as mode strings with no specific value listed.

3.2.3 Client-side application development

A major design decision was made early on to utilize Actionscript 3 (AS3) for development of clients for the game and the media services. This was done because many facilities desirable for game development were either planned or released for AS3 by Adobe. Many of these facilities such as 3D effects are not possible to access under the older Flash Development Tools such as Flash CS3 using AS2. Some unexpected consequences arose in that many previously developed AS2 classes do not work in AS3.

Development of the around game service development effort was done using Flex3 which was a more programmatically conventional IDE. (Based on the Eclipse IDE)

One benefit from using a widely deployed technology such as Flash is that there are many available clients. A search for various applications (mainly open source software) was begun to find FMS clients that we could use as models for our streaming media services. These include but are not limited to:

1. one way video broadcast and multi-way (p2p) video conferencing [18]
2. shared online whiteboard (drawing and application sharing) [19]
3. online chat and IM services [20]
4. video stream recorder and player [21]
5. shared object video conferencing [22]
6. other Red5 media server applications [23]

Our initial reuse efforts began in examinations and installations of various open source or free applications. To simplify work, we excluded Flash development environment-based clients (fla files), MySQL, and legacy AS2-based open source applications because our development was primarily to be in AS3 in the Flex3 development environment.

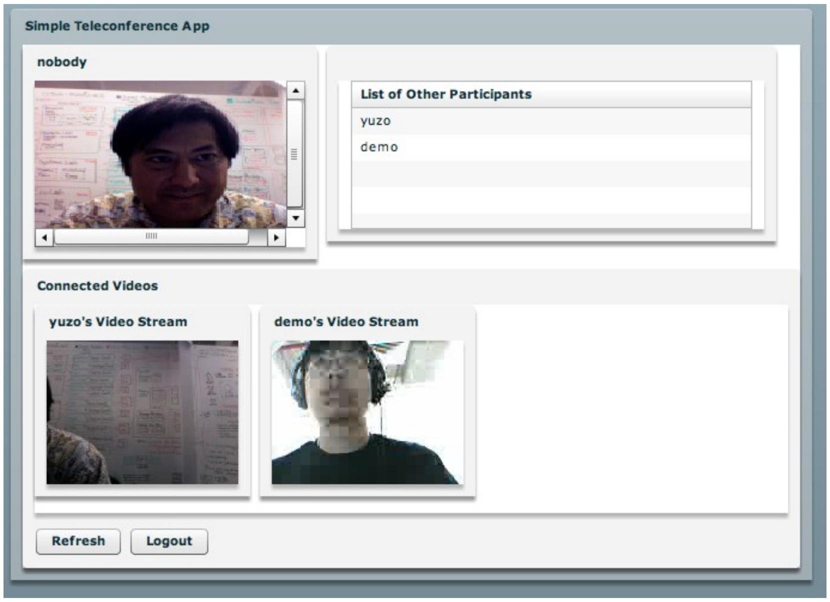


Figure 1. Around-game video teleconferencing application.

Our resultant client applications were built using a mixture of open source and project developed code. These developed clients include a Video Teleconferencing client (see Figure 1.), a Video Player and a Video Recorder. The clients in general did not produce significant server loads, but did cause client machines to have high CPU usage. For instance the Video Teleconferencing client was designed for low bandwidth, low-res video (16kbps per stream), but the loading client-side was: 48% CPU usage (Core 2 Duo Mac OS X 2.4GHz, 4Gb, integrated 1300 graphics) 2 user; and 52% CPU usage 3 user.

3.2.4 Client-side Development Problems

Our initial effort was to drop in Red5 as a FMS replacement in out portal system. To do this, we sought the use of seemingly best reviewed example applications. A freely available example Red5 Teleconference application that used shared objects available to the mail-list for Red5 was initially used as a model for application development. There were positive reviews on the mail-list and other blogs to this application, but direct use of the source code revealed hidden problems.

The principal means of client identification was replicated in both the data stream (shared object) and the video stream (stream name). If a mismatch occurred, then the mechanism of stream identification was affected leading to cases where video streams were connected, but corresponding objects were lost.

Considering the difficulty debugging across the client and server, the approach of eliminating shared object use was adopted since it would limit bandwidth use and avoid seeming data replication.

After initial deployment, the merit of the original shared objects approach became apparent. Shared objects in Flash have built in cross client synchronized data transmission method (synchronize) available. This allows for clients to send lightweight data across to each other concerning overall state. In a Teleconferencing Application there are a few situations where this capability would be useful. For instance, when a participant decides to pause and then later resume streaming. Utilizing a media streaming elements alone would not always work as expected.

Also an application heartbeat mechanism could be setup on the client-side or the server side using the data stream (in the form of shared objects) in order for the application to determine connectivity and deal with overall application consistency situations.

A concern about latency arose in test of the video conference application. Testing with other Flash-based systems (such as dimdim.com) reveal that there is an issue of latency that the current technology has not fully addressed.

3.3 Changes in Architecture

Originally, the architecture of the game infrastructure followed that of a Federated/Hybrid System, where Red5 would replace FMS in a collection of servers providing around game services (see Figure 2.).

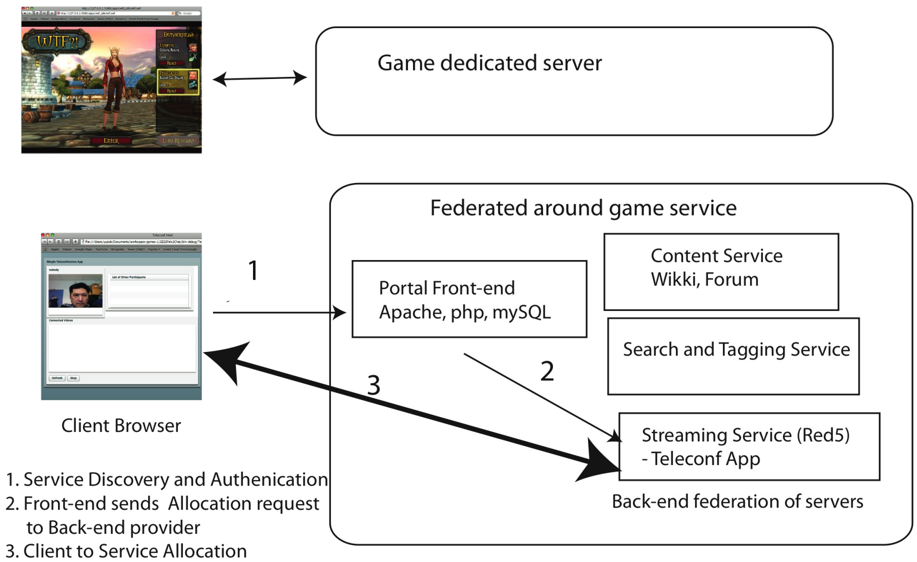


Figure 2. Federated Game Service Architecture.

Architecture Mismatch Issues

Development of around game services in the existing portal's Federated Architecture meant that some constraints were applied to the streaming server. This induced problems because the drop-in components for the Federated Architecture were structured around a set of global concepts that formed the integration basis for development.

The mismatch between expectation of simple server replacement as a low-cost/low-effort transition and the effort required to shoehorn the services was surprising. These mismatches manifested themselves in the following way:

First, utilization of the existing portal identification/authentication services was pursued because it provided a common shared mechanism across the application. This meant that the identity of the allowed/authenticated clients was passed via process environment variables. These parameters reduced to string and integer, so it is possible for a clever hacker to bypass the checking mechanism and hijack the services. But because this service was common between all other services and essential as-is for other components in the system, a reconstruction of the security model (to a stronger certificate-based system) was not possible as it would affect all other components in the system.

Second, while our development plan called for use of AS3 as the base programming language for both the game and the around game services, the wide spread use of the previous version of the language (AS2) caused problems. Notably the event model for the language changed therefore mechanisms commonly used for message passing became obsolete or would not function properly when used or converted to the new base language. This limited the amount of reuse possible with existing code bases.

Third, cross application tools became dependant on specific fix mechanisms. For instance, the Google Analytics tool provides events that can be accessed by a number of programming languages. For Flex3's AS3 complier, these events were often misprocessed leading to null values, while the same event would be accessible by other components built in other languages with correct values. A fix was applied that worked under specific conditions and compiler versions. Similar issues arose with fixes for bugs in the Flash player itself.

Fourth, debugging across multiple process and servers can be problematic. For instance, a particular client or server may depend on a particular ordering of events. This ordering may be implicit in the documentation e.g., the Video Teleconference connection sequence to Red5. The possibilities for a bug could be spread across different servers operating non-deterministically since requests and responses can occur asynchronously. Debug tools for such situations are not as widely known or as widely used as language specific debugging tools.

Unified Architecture

During the development, use of Red5 alone in a Unified Architecture to support the needs of the single-user WTF?! game and the around game streaming services was pursued. This simplified the problems of matching Red5 and the supporting software (written mainly in AS3) to the server environment. Because the multiplayer engine development was delayed, the game service infrastructure for single player use was supported by the simple HTTP transport mechanisms provided by the servlet container application (Jetty). This allowed for a two-level connection mechanism to be employed. Simple resource services were provided through the HTTP service, while streaming was provided through the RTMP service both of which are available under Red5 (see Figure 3.).

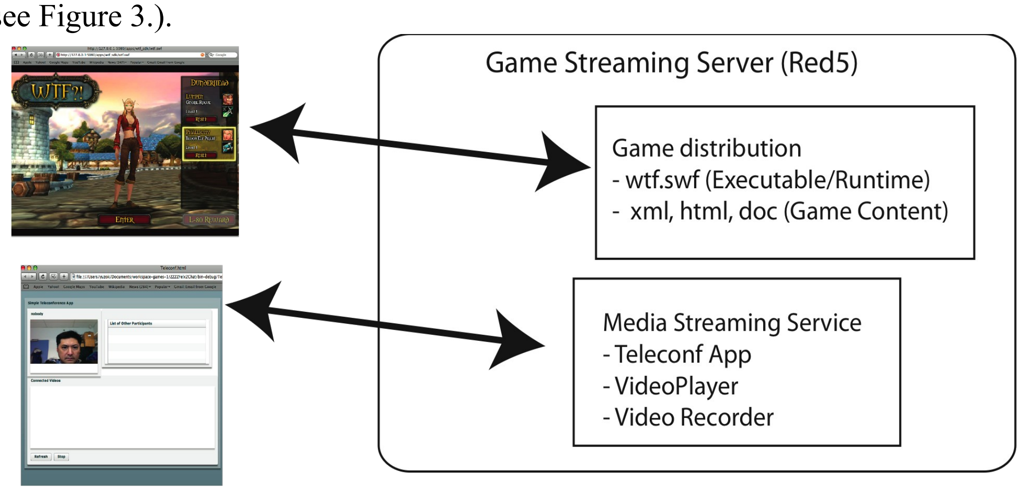


Figure 3. Unified Architecture.

Future Game Streaming Unified Architecture

While Red5 is purpose built in Java to replace functions in FMS, it could be altered to support game infrastructure needs. The servlet containers could be modified to include application code suitable for the data streaming aspects of multiplayer Flash-based games. There are indications from other projects that indicate a combined server could be possible. Project Darkstar [7] is an open source, Java-based game engine. Concepts and capabilities in this project could conceivably be combined with Red5 to from a system with two halves – a game half and a media streaming half. The commercial *SmartFox* server takes this approach and has had success.

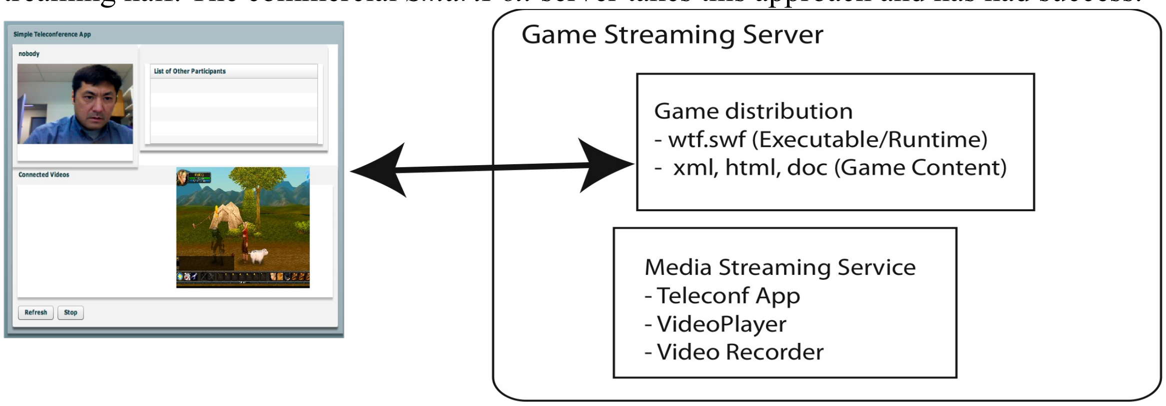


Figure 4. Future Unified Game Server Streaming Architecture.

While a server like *SmartFox* has presented unified services within a single deployment, the design of the server may induce constraints on the design of a multiplayer game. If a novel infrastructure need arises, and a novel new protocol to address it is developed, the possibility of implementing it in a server may be limited by the mechanisms employed in the design of the server. For instance, a new version of the RTMP protocol may be exploitable for multiplayer games, but until all crucial aspects of the new protocol are hard coded into the server, the developer is limited to what services are available.

Integration of multiple services into a single channel or through a single protocol may be difficult. Because some protocols like HTTP are connection oriented, they may lack the mechanisms to provide timely and consistent data transfer needed for games. Research in game specific protocols shows that some underlying constraints of the underlying REST architectural style may have to be altered to provide a good gaming experience. REST is the architectural style behind the HTTP/1.1 protocol [16]. REST does not address concerns raised in scaling multiplayer game engines, but enlightens how to induce benefits by enumerating how sets of beneficial properties were combined to produce the HTTP protocol. A future unified server, perhaps based on CREST technology [17] should either support multiple streaming protocols to supply services as needed, or it may support a single grand protocol that would support all aspects of game interaction and all aspects of data and media streaming. (see Figure 4.)

Future Game P2P Architecture

The communication mechanism for games is dependent on the constraints of the system resources and the overall goals of the game play. This has led to refinements of data transfer protocols to provide the necessary infrastructure for certain classes of games e.g., multi-player. [24]

Future game architectures could apply constraints critical to their system operation and enhance properties that would beneficial expressed in a non-traditional P2P architecture. For instance, many current file sharing protocol systems proceed under the assumption that total bandwidth utilization is desirable to transfer files. Bandwidth limiting protocol systems such as Skype [26] and Joost [27] limit bandwidth utilization to provide the necessary media service (audio and/or video) even though they have the same underlying P2P technology as bandwidth consuming file sharing protocols. This shift in bandwidth consumptions allows an existing P2P technology to provide a service most often deployed in a client/server architecture.

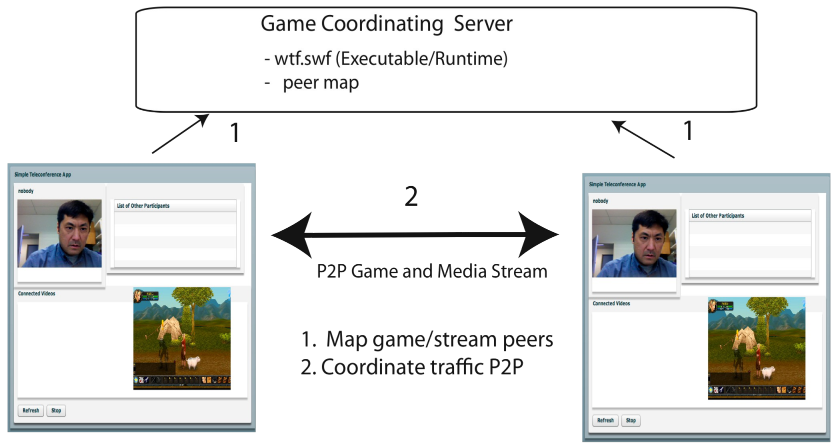


Figure 5. Future P2P Architecture.

It is conceivable that P2P connection pathways may allow for the efficient construction of this type of game architecture for our streaming use in the future. (see figure 5.)

4. DISCUSSION

Services for game infrastructure have often been split between a server supporting game play only and a set of servers to supply other around game services. This often leads to complex hybrid architectures with many individualized services being federated together. While federation can have beneficial aspects, such as collecting best-of-breed services together, there can be complexities in installation and maintenance. Unexpected component mismatches in the architecture can occur and solving such mismatches can take more resources than would initially be expected. Drop-in replacement of Red5 for FMS did not occur smoothly in our federated system. This was due to evolution in code-base to be supported and subtle difference in the semantics of client server interaction.

A unified streaming architecture offers benefits for small scale deployments and reduces the mismatching of desired services with provided services.

In our development, Flash was the base technology used for both the game and the around game services. In Flash a distinction is made in “remoting” (formats for data which are distributed or streamed) and of media streams. This led to our two-level unified architecture, where some pathways follow HTTP connections, while others follow RTMP pathways. For an application such as Videoconferencing both data a media streams complement each other, but can often be confused as to what purposes they serve. Because both can be considered instances of streams at different data rates, a framework to utilize them in one structure would be beneficial for future enhancements towards unified and P2P architectures.

A tension that arises in service development. On the one hand, a simple use of services-as-provided by a component can be used to manufacture services in federated architectures. An example of such as use of Red5 is in the Drupal module which uses the existent demo applications bundled with Red5 to provide services through a thing executable swf layer. Such a system would be exposed to potential security holes in the Red5 demos and to being hijacked for use by clever programmers. On the other hand, more complex and secure services could be developed which would require more of a commitment to the development of applications on the server-side in Java. This could limit the flexibility of the overall game infrastructure by adding a technology which may run counter to the game's needs.

The needs of audio/video streams vary depending on factors such as type of application – video playback allows for bandwidth checks to buffer data which teleconference puts a premium on low latency. Game data exchange for multiplayer games put premiums on latency and data size as well as synchronization. Streaming is just one part of an overall server application for games. Services may be layered on top of streaming technologies, or may co-exist in a federation of other services.

A long term problem of application latency for teleconference is troublesome. On Red5, an observered 1000ms latency was seen between test machine pinged with 25ms response, yet the latency of skype on the same machines was only 130ms. While only an order or magnitude difference, the result on local teleconferences was perceptible.

5. CONCLUSION

There exists no one game server infrastructure or architecture that can address all concerns in all possible game development. Each game engine is typically designed to optimize the play of one particular game. Similarly, there is no one universal streaming mechanism.

Aspects of game data exchange, which is the core interaction mechanism enabling multiplayer games, exhibit characteristics addressed in streaming services, e.g., how to move data in a timely fashion, how to connect multiple users, etc. The movement of game data can be seen as an example of a stream. The customization of game servers to optimize data exchange can lead them to the point where new capabilities such as video data exchange are not feasible. If games are to increase bandwidth usage to provide greater user experience, then the use of streaming servers could help simplify the servers that are constructed for use in a game system.

While Red5 provides a workable solution to our current needs, it by itself is not a total answer to concerns that arise from multiplayer game development especially scaling. Management of users persistent data, use of unconventional data transmission methods are large scale concerns not applicable to our investigation nor a direct concern of the streaming server we examined.
It is likely that others would be interested in incorporating more streaming services into their game architectures. Such systems should think carefully about how streaming should be designed into their systems.

Because of efforts in open source game engines such as Project Darkstar and in Flash Media Servers such as Red5, it is conceivable that a Flash-base game engine could be developed. Issues such as video streaming latency must be overcome as alternate means such as Skype offer better performance in Teleconference.

5. ACKNOWLEDGMENTS

Robert Nideffer and Alex Szeto at the UCI Game Culture and Technology Laboratory made frequent comments, requirements statements, and critiques of the work described here. However, they should not be held liable for such contributions to our effort. Support for the research described in this report comes from grants #0534771 and #0808783 from the National Science Foundation, Intel, and the Daegu Global R&D Collaboration Center, Daegu, South Korea. However, no endorsement is implied.

6. REFERENCES

[1] Nideffer, R., Wang, A.I., and Szeto, A, 2008. *unexceptional.net*, submitted for publication. Also

[2] Scacchi, W., Nideffer, R. and J. Adams, 2008. A Collaborative Science Learning Game Environment for Informal Science Education: *DinoQuest Online*, in IFIP International Federation for Information Processing, Vol. 279; *New Frontiers for Entertainment Computing*; P. Ciancarini, R. Nakatsu, M. Rauterberg, M. Roccetti (Eds.); Boston: Springer, 71–82. Also see *Dino Quest Online*, at http://www.DQOnline.org

[3] *WTF?!* Version 1.0*,* 2008. http://www.aoedipus.net

[4] Brown, C. and Scacchi, W., 2007. Modeling and Navigating a Simulated Semiconductor Fabrication Laboratory using a Game Engine, http://www.ics.uci.edu/~wscacchi/GameLab/FactoryVisualization/FabLab-Draft-Scacchi-15Oct07.pdf Also see, *FabLab Game Demo Reel#3*, 2008. http://www.ics.uci.edu/~wscacchi/GameLab/DemoReels/FabLab-Demo-03.wmv

[5] Alexander, T. (Ed.), 2003. *Massively Multiplayer Game Development*, Charles River Media, Boston, MA.

[6] Levine, D. and Wirt, M, 2004. Interactivity with Scalability: Infrastructure for Multiplayer Games, in Foster, I. And Kesselman, C. (Eds.), *The GRID2: Blueprint for a New Computing Infrastructure*, Morgan Kaufamann, Paolo Alto, CA, 167-178.

[7] Project Darkstar, 2008. http://www.projectdarkstar.com/

[8] Kumar, S., Chhugani, J., Kim, C., Kim, D., Nguyen, Dubey, P., Bienia, C. and Kim Y., 2008. *Second Life* and the New Generation of Virtual Worlds, *Computer*, 41(9), 46-53.

[9] OpenSim, 2008. http://www.opensimulator.org

[10] Taylor, T.L. 2005. *Play Between Worlds: Exploring Online Game Culture*, MIT Press, Cambridge, MA.

[11] Nardi, B. and Harris, J. 2006. Strangers and Friends: Collaborative Play in World of Warcraft, *Proc. ACM Conf. Computer Supported Cooperative Work (CSCW'06*), Banff, Canada., ACM Press, 149-158.

[12] Lowood, H. 2006. Storyline, Dance/Music, or PvP? Game Movies and Community Players in World of Warcraft, Games & Culture, 1, 362-382.

[13] Oehlberg, L.; Ducheneaut, N.; Thornton, J. D.; Moore, R. J.; Nickell, E., 2008. SocialTV: designing for distributed, social television viewing. *International Journal of Human-Computer Interaction*. 24(2). 136-154.

[14] Weisz, J.D. And Kiesler, S., 2008. How text and audio chat change the online video experience, *Proc. 1st International Conf. on Designing Interactive User Xperiences for TV and Video*, 9-18, Silicon Valley, CA, ACM Press.

[15] Bass, L., Clements, P., and Kazman, R. 2003. *Software Architecture in Practice*, 2nd Edition, Addison-Wesley, New York.

[16] Fielding, R. and Taylor, R.N., 2002. Principled Design of the Modern Web Architecture, *Trans. On Internet Technology*, 2(2), 115-150.

[17] Erenkrantz, J.R., Gorlick, M., Suryanarayana, G., Taylor, R.N. 2007. From representations to computations: the evolution of web architectures, *Proc. 6th Joint Meeting European Software Engineering Conf. And the ACM SIGSOFT Symposium on the Foundations of Software Engineering*, Dubrovnik, Croatia, 255-264.

[18] Renaun Erickson's FMS Flex Teleconference Application example: http://flex.sys-con.com/node/295379
[19] Sunil Gupta's Whiteboard application: http://sunil-gupta.blogspot.com/2007/04/shared-whiteboard-application-in-red5.html
[20] Chat Collab environment for FMS and MySQL: http://coenraets.org/blog/2006/10/building-collaborative-applications-with-flex-data-services-and-flash-media-server/
[21] Carl Sziebert's Video Stream recorder: http://sziebert.net/posts/server-side-stream-recording-with-red5/
[22] Shared Object-based Red5 Video Conference (Uses Flash development environment): http://lab.urban-sky.net/red5_videoconference/
[23] Other Red5 media server applications: http://osflash.org/pipermail/red5_osflash.org/2007-March/010112.html
[24] Esbensen, D. 2005 The Dark Side of MMORPGs: Infrastructure Gone Mad. *GDC 2005 Proceeding: Online Game Architecture: Back-end Strategies*. http://www.gamasutra.com/view/feature/2242/gdc_2005_proceeding_online_game_.php
[25] Red5 FCS/FMS Migration Guide http://www.joachim-bauch.de/tutorials/red5/MigrationGuide.txt
[26] Skype 2008. http://www.skype.com/
[27] Joost 2007. http://www.joost.com/

Video Capture Tools

Yuzo Kanomata

Goals

- To support multiple platform use
- To allow for quick, near real-time production
- To find capture tool(s) to enable recording of:
  - Web-based conferencing
  - Virtual Worlds/Game expereiences
  - Streaming presentations

Platform dependencies

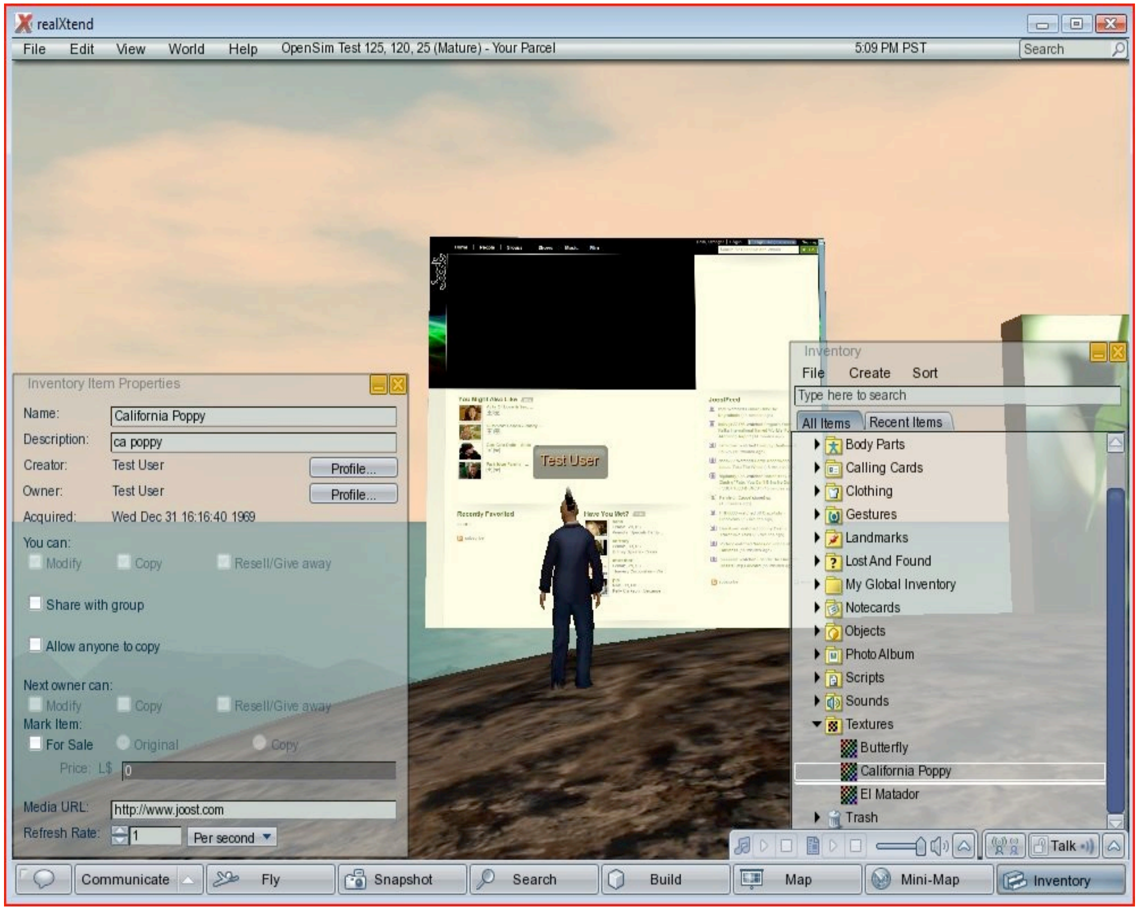
- OS system presents screen information in platform specific mechanisms (Windows, Mac, X11, Gnome, KDE, etc)
- Partial mechanisms for cross-platform exist (Java robots) but good frameworks to utilize images do not exist.

Recording Tools Evaluated

- PC-based
  - Camtasia (Studio and Relay)
  - FRAPS
- Mac-based
  - Camtasia Relay
  - Snapz Pro
- Cross-plaform
  - Java-based

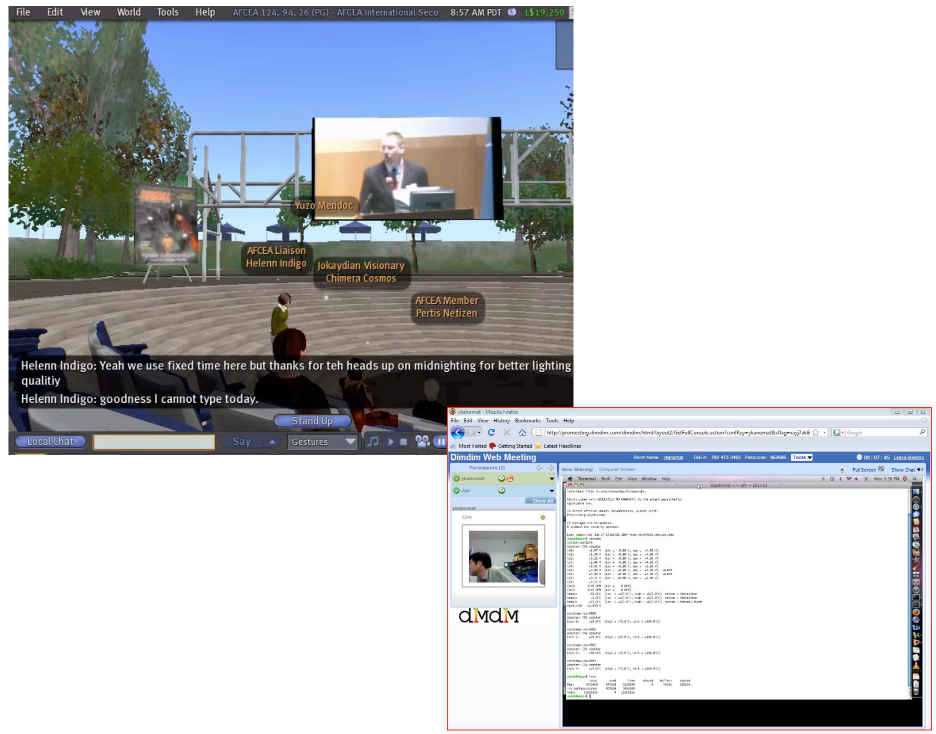
Camtasia Studio

- Works well on XP, Vista is problematic.
- Can inject events/ metadata during capture-time or post production, but these are packages as swf helper apps for later viewing (flv content type).
- Requires substantial GPU to operate cleanly



Camtasia Relay

- Client/Server-based
- Mac and PC
- Problems with driver usage, options on platforms vary
- Problems in sound recording

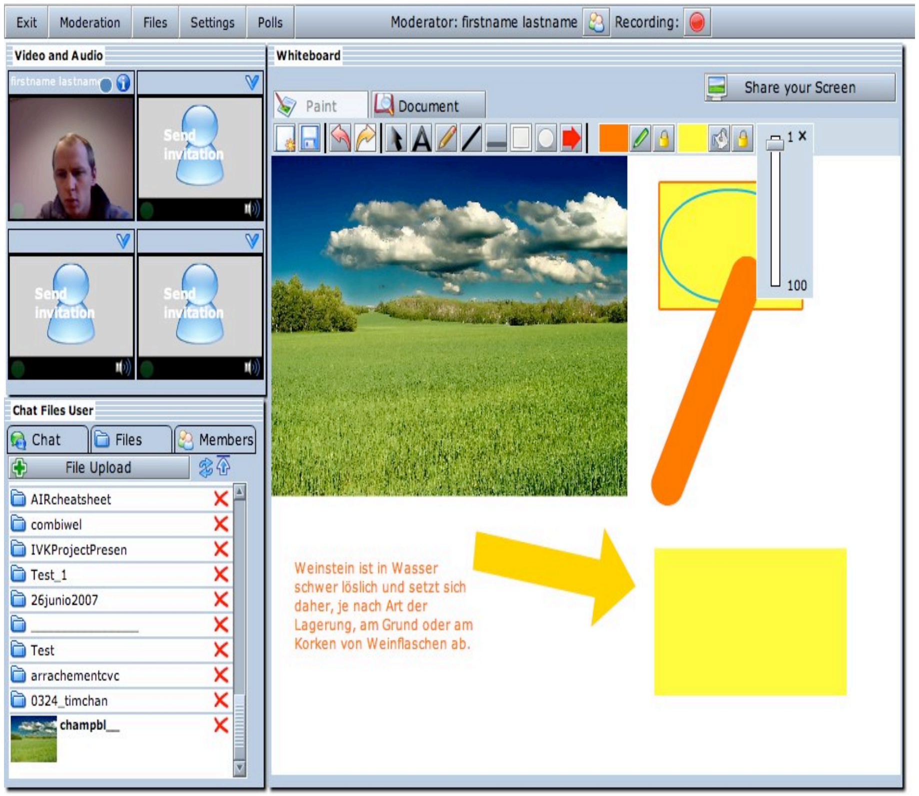


Fraps

- Only Windows-based.
- Only Application-based, no browser capture capability
- Requires GPU

Java Robots

- Takes advantage of a Java abstraction.
- Must program your own system
- Output formats may be a problem
- Cannot access some application frames
- Many applications cannot be captured



Snapz Pro

- Mac only
- Long post capture processing for MPEG-4, but resolution much better than faster Sorensen Sparc codec

Tools not yet evaluated

- Adobe Captivate 3
- Jing
- BTV
- iShow
- Uvcn2swf
- ScreenographyScreen
- MimicScreen
- RecordBlip.tv
- Screencast
- ViewletBuilder

Camtasia Specific XP/ Vista issues

- Camtasia Studio
  - Vista usage will require a separate codec Dvix to run well
- Camtasia Relay
  - No codec selectable, limited audio selectors.

Good Formats for Use

- To support a wide array of viewing options, flv and mpeg-4 are the most beneficial
  - On-site delivery possible through media and streaming servers (FMS, Red5, etc.)
  - Off-site services accept these formats (YouTube, Akamai, etc.)
- Editors accept these formats
  - Tools to add metadata exist for widely used formats

Common Problems

- Changing parameters (screen size, audio capture target) during a capture not supported.
- Recorded formats may be platform specific
- Recording of other data streams such as text chat are indirect
- Real-time tagging not supported

Limitations

- Capture trades off space/time vs. resolution/fps
- Not all tagging/mark-ups are currently supported (e.g., Screencast.com cannot handle all screen captures done by the front-end tool.)
- Real-time production limited for the present, post-production activities can be unbounded.
- Tools and techniques are in flux

Virtual Worlds: Capturing Videos

Yuzo Kanomata

Problem

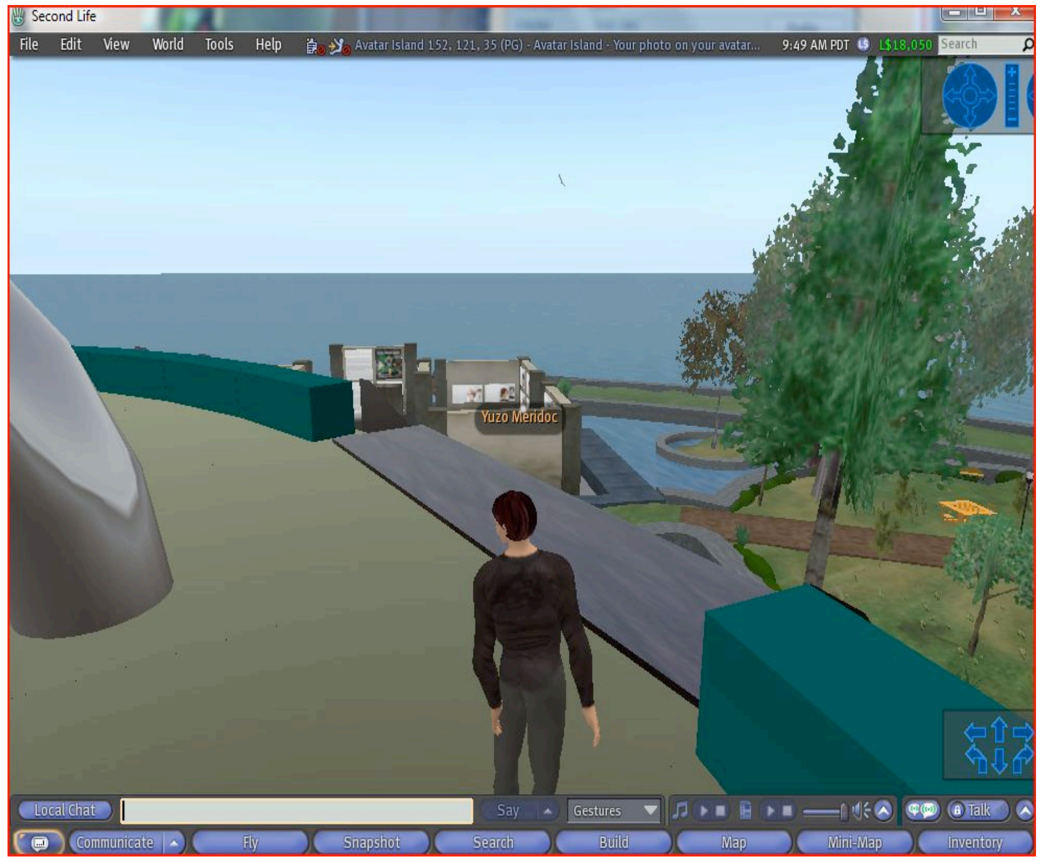
- Virtual Worlds are presented as venues for collaborations (meetings, presentations) yet they often do not natively support capture capabilities.
- Tools to capture virtual worlds differ per user platform and can only partially collect information in an accessible way for later use.
- What point of view to capture can be an issue.
- No available text capture mechanisms.

Using Capture Tools

- FRAPS
  - Used for Game frame capture
  - Windows only
  - Cannot be used to capture a browser or multiple windows
- Camtasia Studio
  - Windows only
  - Very troublesome on Vista 64
- Snapz Pro
  - Mac only

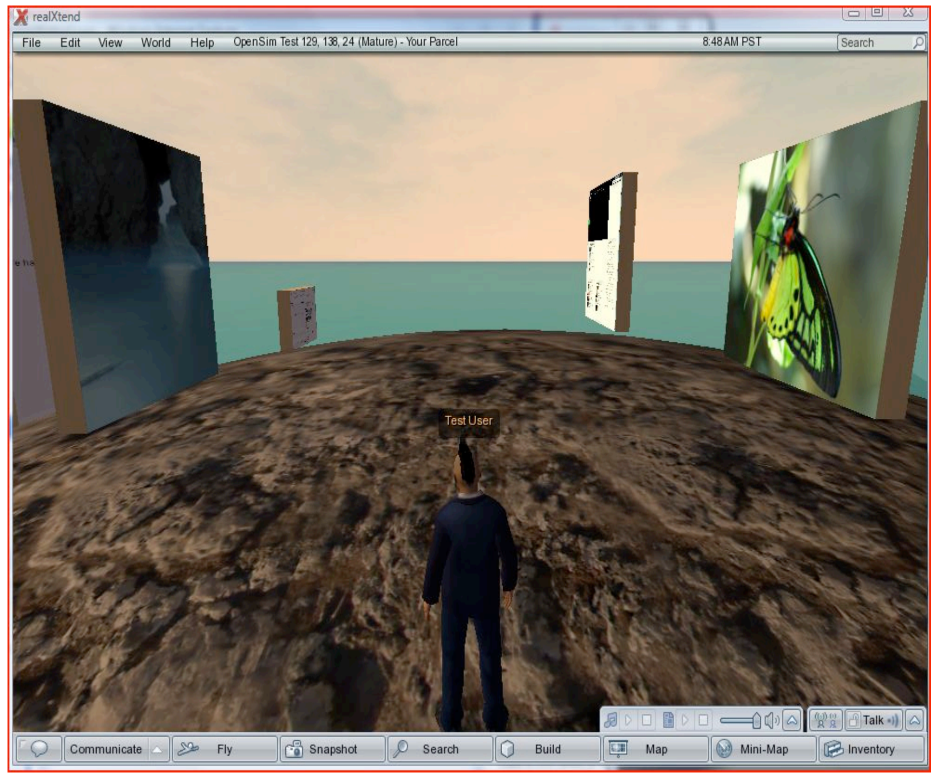
FRAPS Capture: SecondLife

- FRAPS can only be used on applications like the SL client.
- Post-production necessary to integrate and convert to formats such as mpeg-4 and flv



Camtasia Capture: RealXtend

- PC-based
- Packaged with editor
- Supports multiple formats
- Troublesome on Vista
  - Requires use of Dvix codec to be stable
  - Sound recording options limited



Snapz Pro Capture : EON Colosium Demo through Citrix

- Mac-based recording
  - No native record capability for EON VW
- Recordable to multiple codecs

Common problems in Tools

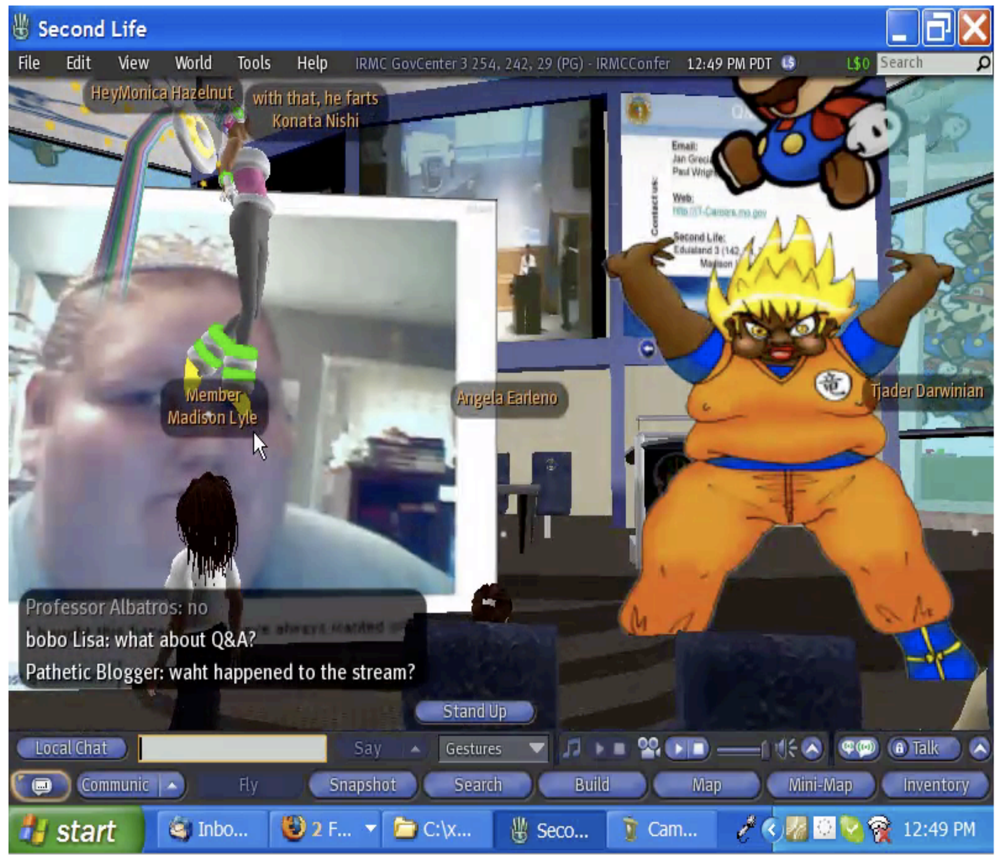
- All tools cannot dynamically reconfigure recording area.
- Access to streams such as audio or text not always available.
- On-the-fly frame annotation very limited
- What/who's view is the most useful?
  - Switching views not always possible

Benefits to recording

- Ability to capture events unexpected by the group
- Ability to do post-event analysis
- Ability to capture otherwise ethereal events
- Ability to archive events

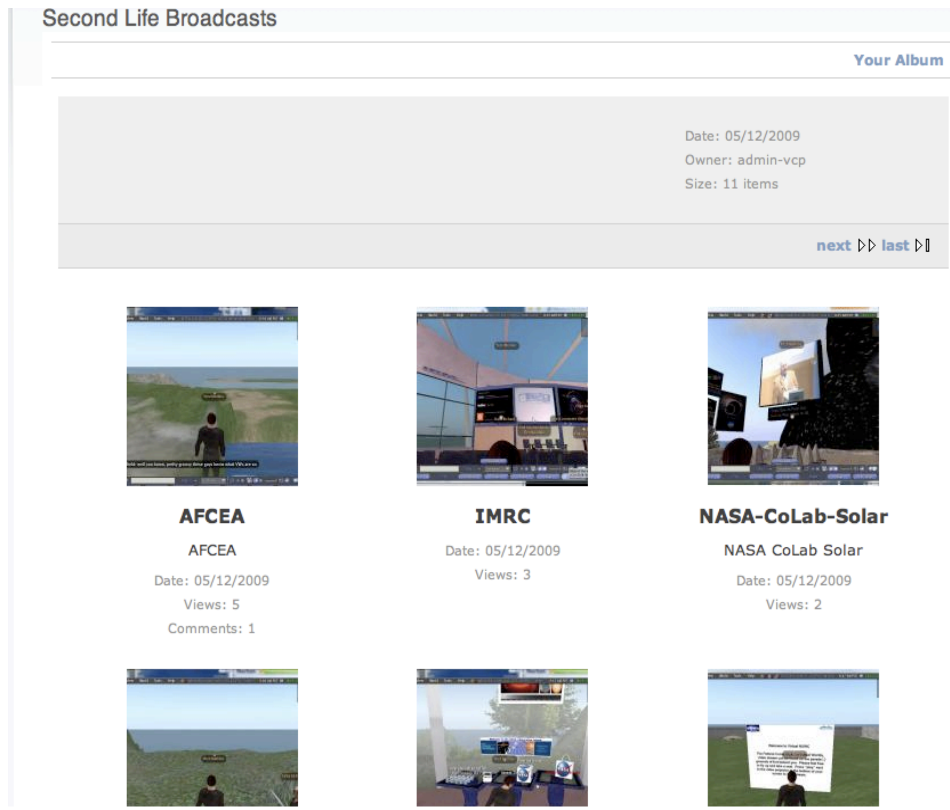
Camtasia: Observed Event Hijack SecondLife

- A large presentation in SL to an audience of 50+ observers
- Presenter was not aware of hijack
- SL Security problem
  - Open systems can allow hackers
  - Gatekeepers can be overwhelmed

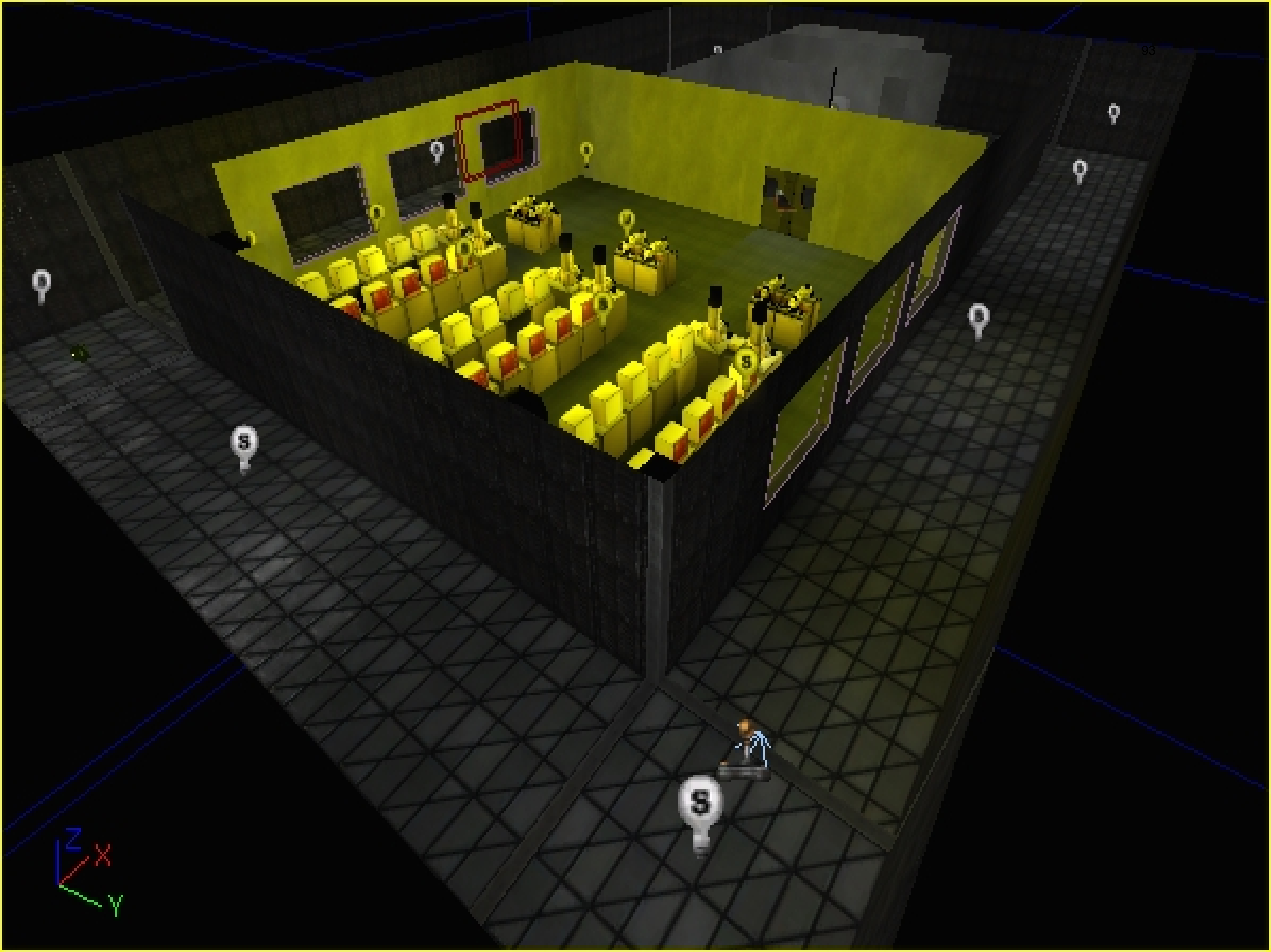


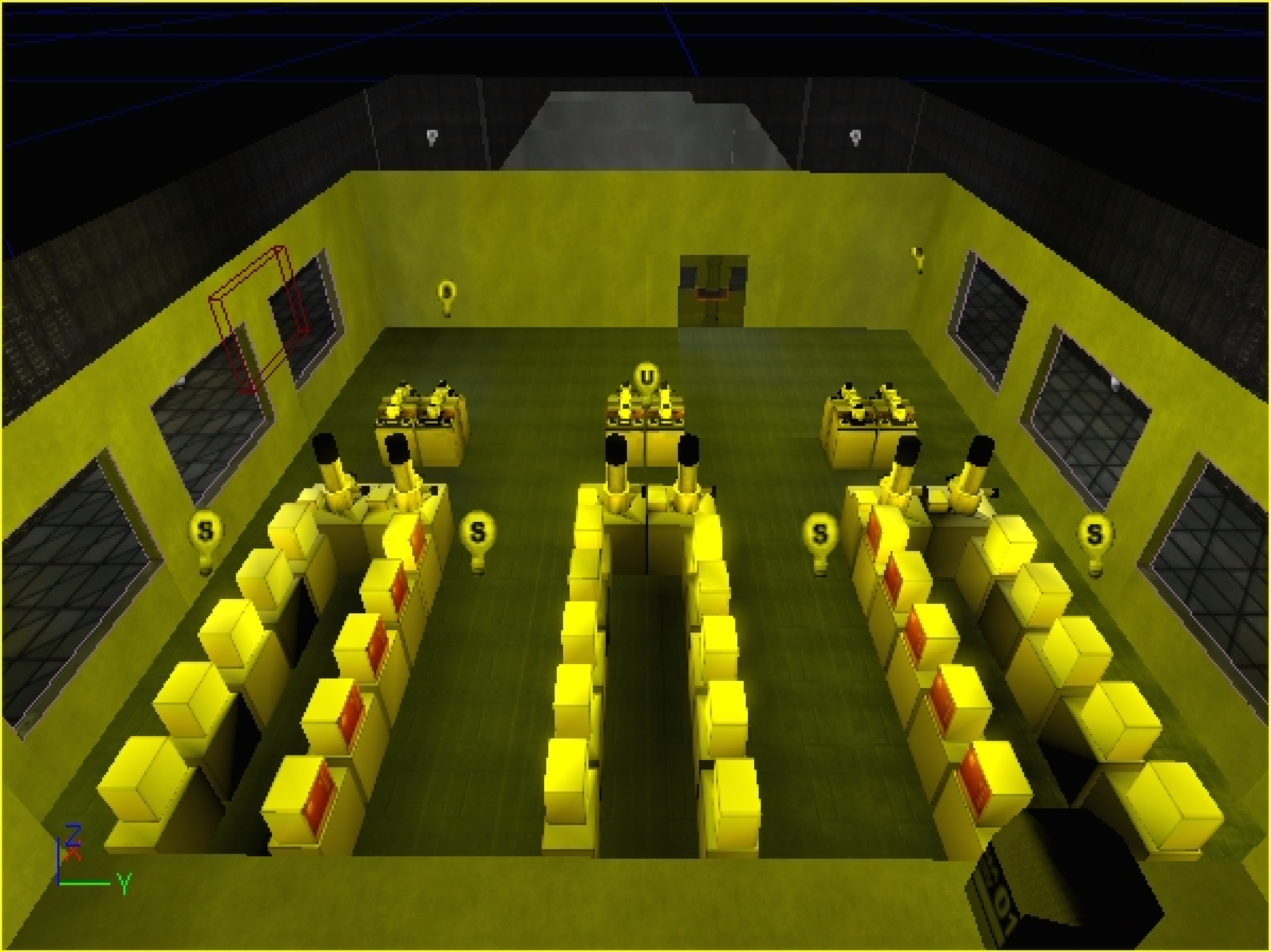
Virtual Collab Portal: VW recording area

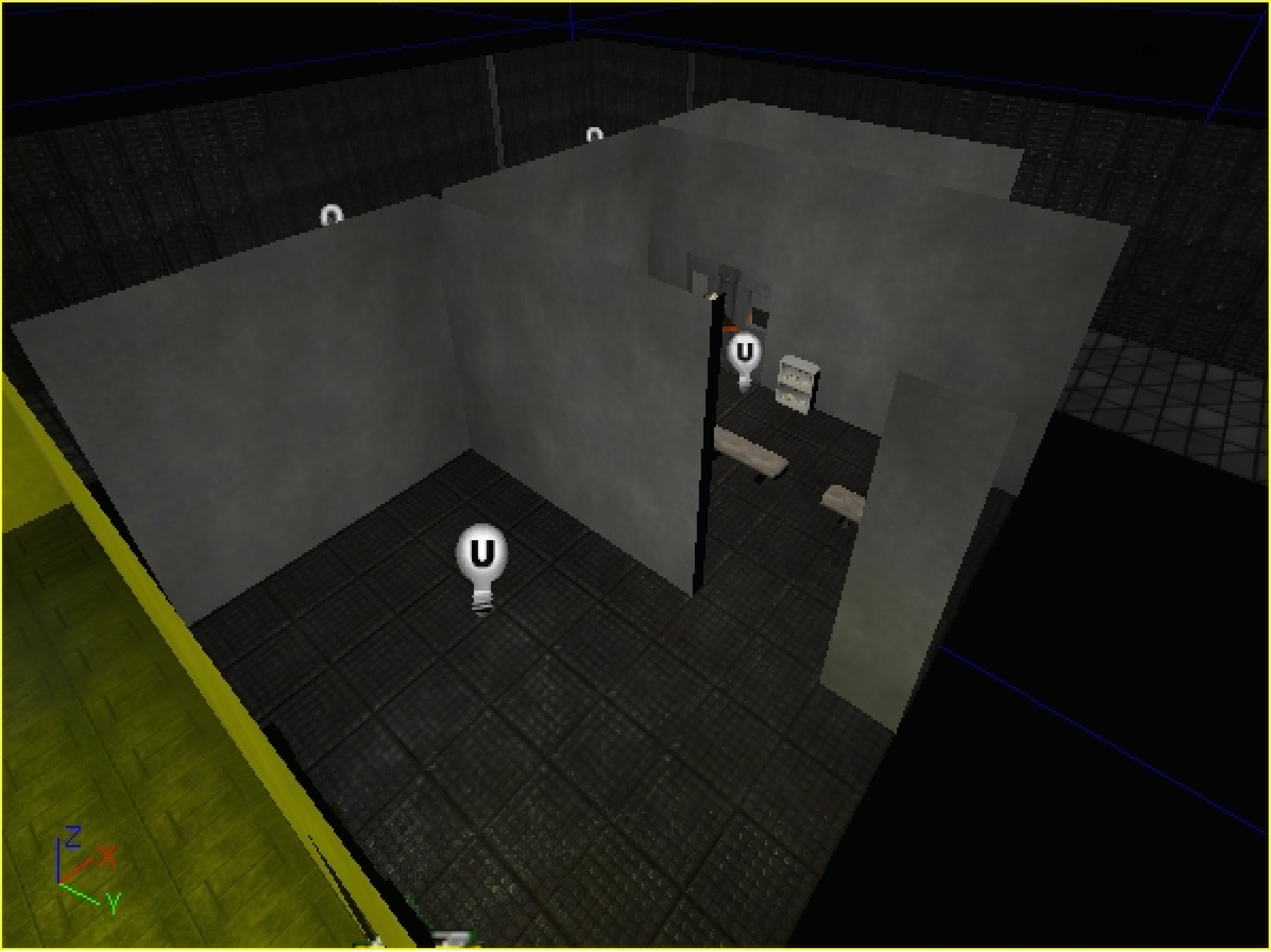
- Example of recordings used by collaborators to do analysis and research
- Videos + Metadata
  - Comments
  - Tags
  - References



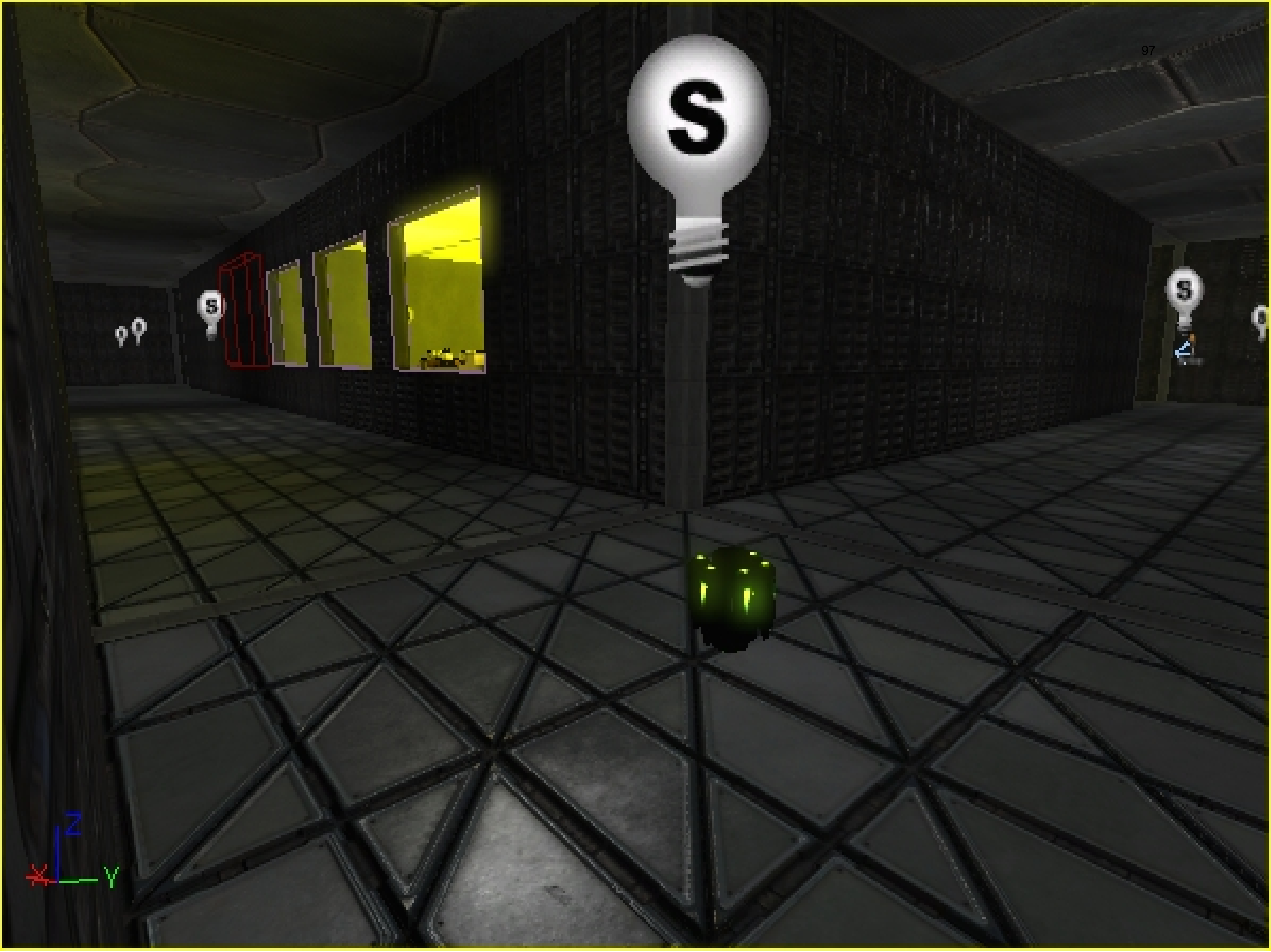
Serious Games R&D

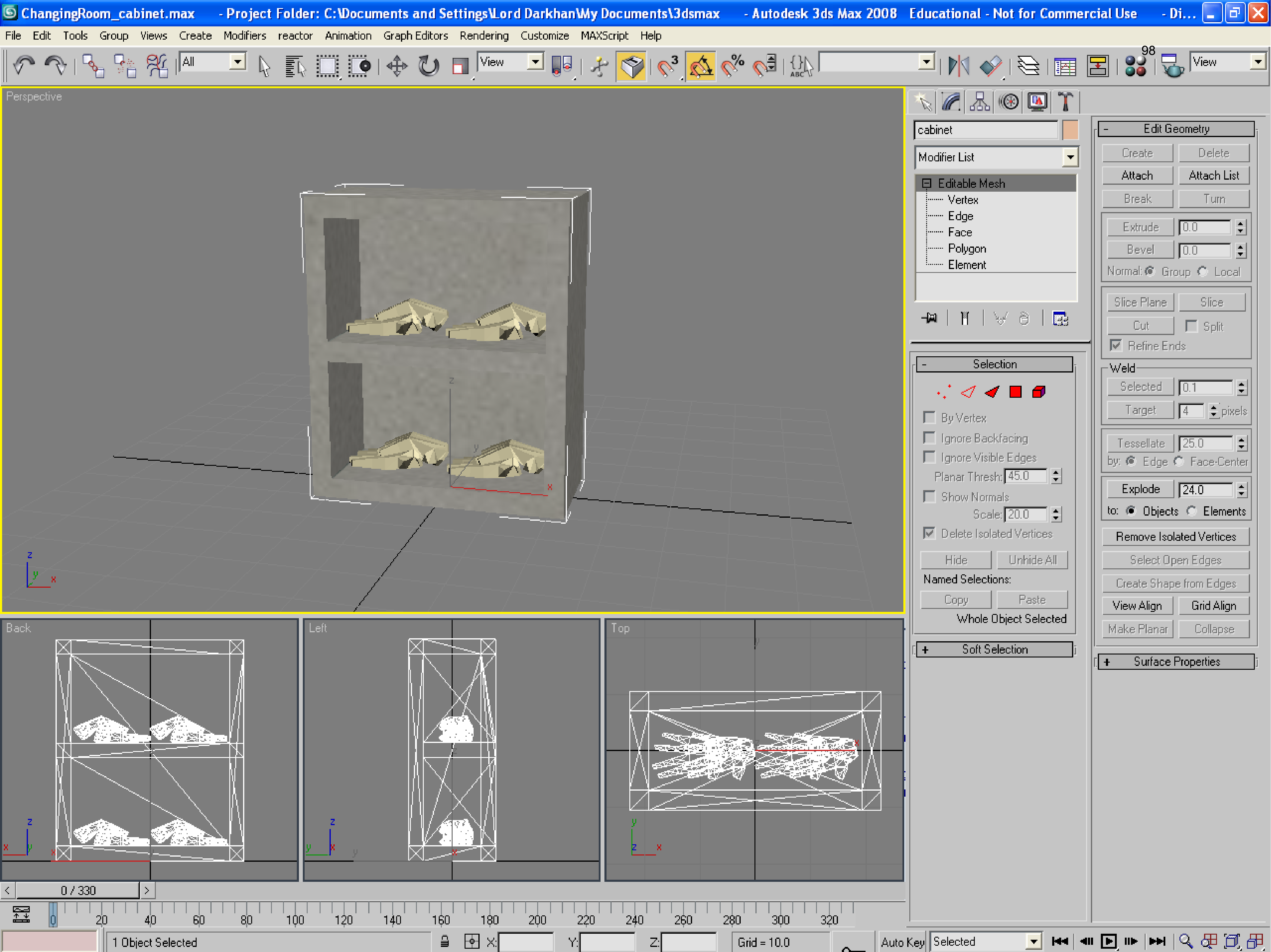

Z
X
Y


U
S
S
S
S
Z
X
Y


Z
X
Y


U
Z
Y
X


S
Z
X
Y


ChangingRoom_cabinet.max - Project Folder: C:\Documents and Settings\Lord Darkhan\My Documents\3dsmax - Autodesk 3ds Max 2008 Educational - Not for Commercial Use - Di...
File Edit Tools Group Views Create Modifiers reactor Animation Graph Editors Rendering Customize MAXScript Help
All
View
View
Perspective
cabinet
Modifier List
Editable Mesh
Vertex
Edge
Face
Polygon
Element
Selection
By Vertex
Ignore Backfacing
Ignore Visible Edges
Planar Thresh: 45.0
Show Normals
Scale: 20.0
Delete Isolated Vertices
Hide
Unhide All
Named Selections:
Copy
Paste
Whole Object Selected
Soft Selection
Edit Geometry
Create
Delete
Attach
Attach List
Break
Turn
Extrude 0.0
Bevel 0.0
Normal: Group Local
Slice Plane
Slice
Cut
Split
Refine Ends
Weld
Selected 0.1
Target 4 pixels
Tessellate 25.0
by: Edge Face-Center
Explode 24.0
to: Objects Elements
Remove Isolated Vertices
Select Open Edges
Create Shape from Edges
View Align
Grid Align
Make Planar
Collapse
Surface Properties
Back
Left
Top
0 / 330
0 20 40 60 80 100 120 140 160 180 200 220 240 260 280 300 320
1 Object Selected
X:
Y:
Z:
Grid = 10.0
Auto Key
Selected

Storyboard for a Computer-Based Analyst (CBA) Training Game:

A Serious Game for Enterprise Resource Planning

Craig Brown and Walt Scacchi
Institute for Software Research
University of California, Irvine
Fall 2008

Summary

This document represents a storyboard (in draft form) for a new serious game for training business analysts in the use of an enterprise resource planning system (ERP), like the ones offered by SAP, Oracle, and other companies.

It is envisioned as a generic effort, rather than one focusing on a specific business enterprise, though our effort draws from experience and knowledge of large enterprise practices surrounding the use of a modern ERP and other computer-based systems for order fulfillment, purchasing, and procurement [Scacchi 1991].

The development of this game focused on rapid development, short schedule (measured in weeks), low cost (including use of low cost game development tools), and development by small team with little prior experience in computer game development. This led to design choices such as 2D game play, layout, and simple game art assets. Game play is situated within an office in a building separate from the ERP system. The game player, acting as a CBA, must travel to and from his/her desk where there is a phone to handle incoming calls/queries from customers, and a desktop workstation for placing orders into the ERP system through the player's workstation user interface. The player must sometimes travel to another location within the building to interact with a different, shared interface to the ERP system.

In the storyboard that follows, the ERP system is designated as the "HUB" to signify its central role in an enterprise information workflow for a computer-based analyst who must use the HUB to manage, track, and fulfill external customer orders for manufactured goods. Also, to avoid confusion, example names for "Customers" used in the storyboard have been redacted, so that any resemblance to a real-world enterprise is unintentional or purely coincidental.

W. Scacchi, Redesigning Contracted Service Procurement for Internet-based Electronic Commerce: A Case Study, *J. Information Technology and Management*, 2(3), 313-334, 2001.

CBA related events:

Demand Forecast Receive and judge Customer Forecast in collaboration with customer.
Weekly, Mon-Wed

**Allocation Enter and disperse Geo allocation qtys down to customer allocations - tied to hub allocations (Monthly)
NOT FOR THE CBAs, JUST RECEIVED
Monthly, last Fri of the month

Allocation Escalation Geos, MMBP, Division will collaborate to try and close any escalated deltas between judged forecast and allocation
Daily, Mon-Fri

Response to Forecast Analyze judged forecast vs. allocation data, identify and action (where applicable) any allocation adjustments, escalate any remainder
Send RTF to customer
Weekly, As soon as customer 2 allocations are finalized, EOW latest

SafetyStockAllocation Calculate the available Hub Safety Stock for current week+1
Allocate Safety Stock to customer specific, Free for Some (FFS) and Free for None (FFN) buckets.
Daily, before Daily BOH

Order Fulfilment Monitor and manage customer pull orders from the hub.
Review GATP check results and process orders for which full amount is not confirmed
Daily

Daily BOH Communicate available remaining customer allocations + available dedicated safety stock for next day order pulls
Daily, by 2:30
Consider following timeline:

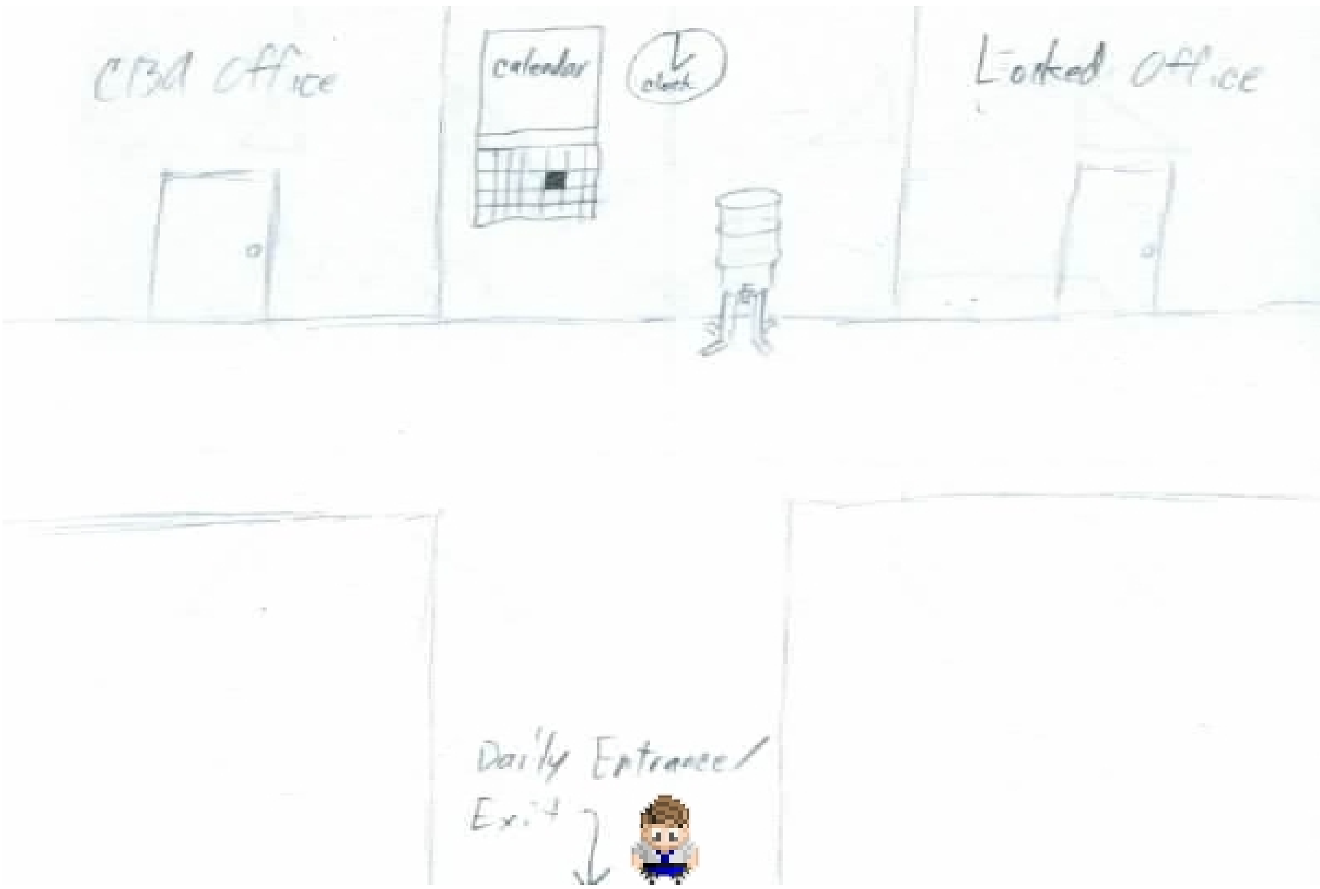
--Daily--*
Make sure HUB has enough Stock for current week (SafteyStockAllocation)
Tell customer how many pulls are available (DailyBOH)
Monitor customer pulls, advise if too many/few are made (OrderFulfilment)

--Weekly--
Early Week SOME_NUMBER of DemandForcasts arrive, input into HUB (DemandForcast)
Mid Week SOME_NUMBER of AllocationCommits come back in different order, fix number in HUB (AllocationEscalation)
Fri Make sure to send RTFs to customers, telling them what will be available (ResponseToForecast)
-Difficulty by week-
Week1 Mon-Tue, Wed-Thu
Week2 Mon-Wed, Tue-Thu
Week3 Mon-Wed, Tue-Fri!!!

--Monthly--
Last Mon-Thu Check which customers have regularly taken fewer than requested
Last Fri Lower DemandForecasts as appropriate for customers who take too few

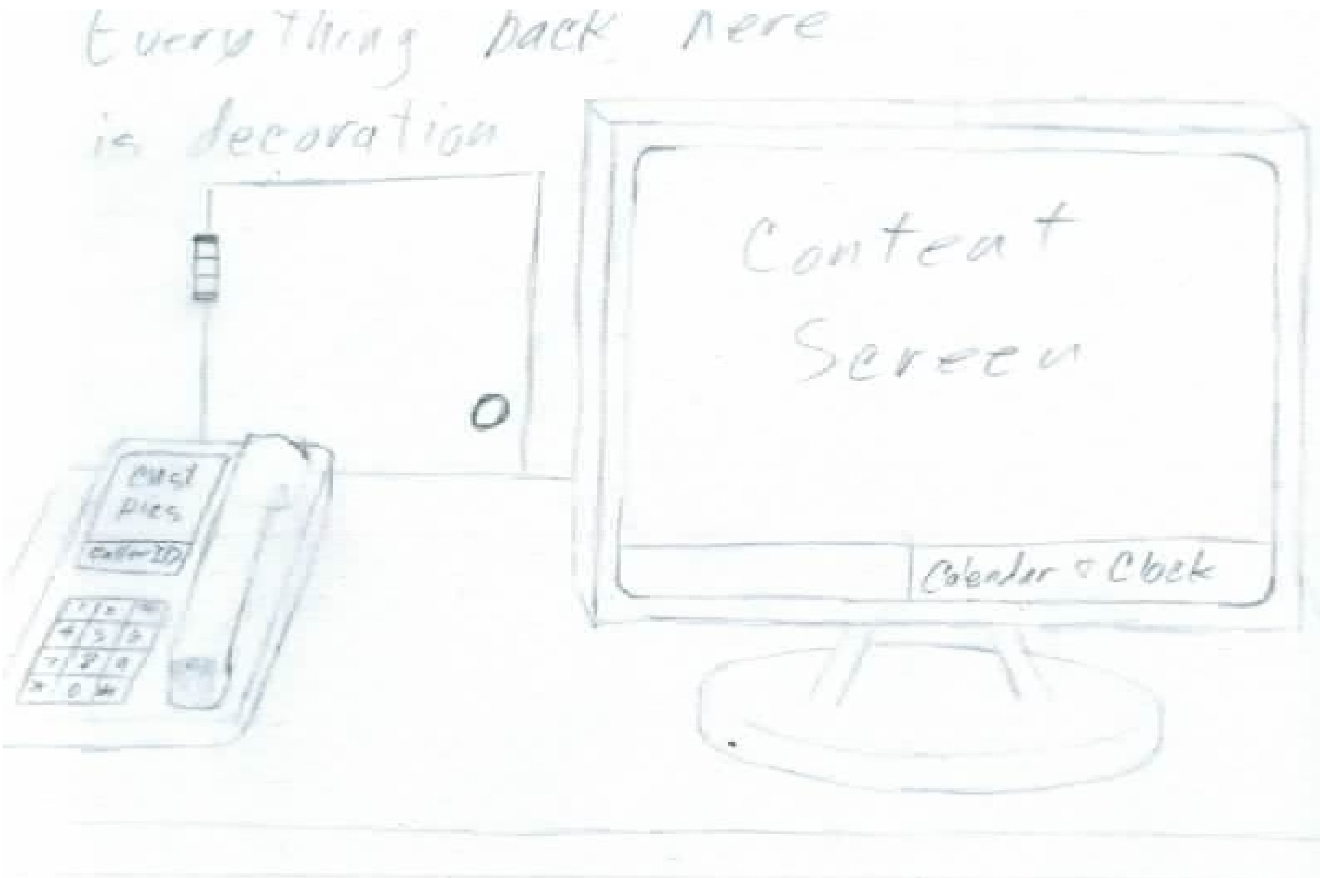
*Daily tasks beging Week2 in game, so CBA doesn't have to pick up from before the game started

This is a where the CBA enters the building at the beginning of each day. This slide (and a few others) will be needed multiple times, but will only appear once. They will be described as “Movement slides”

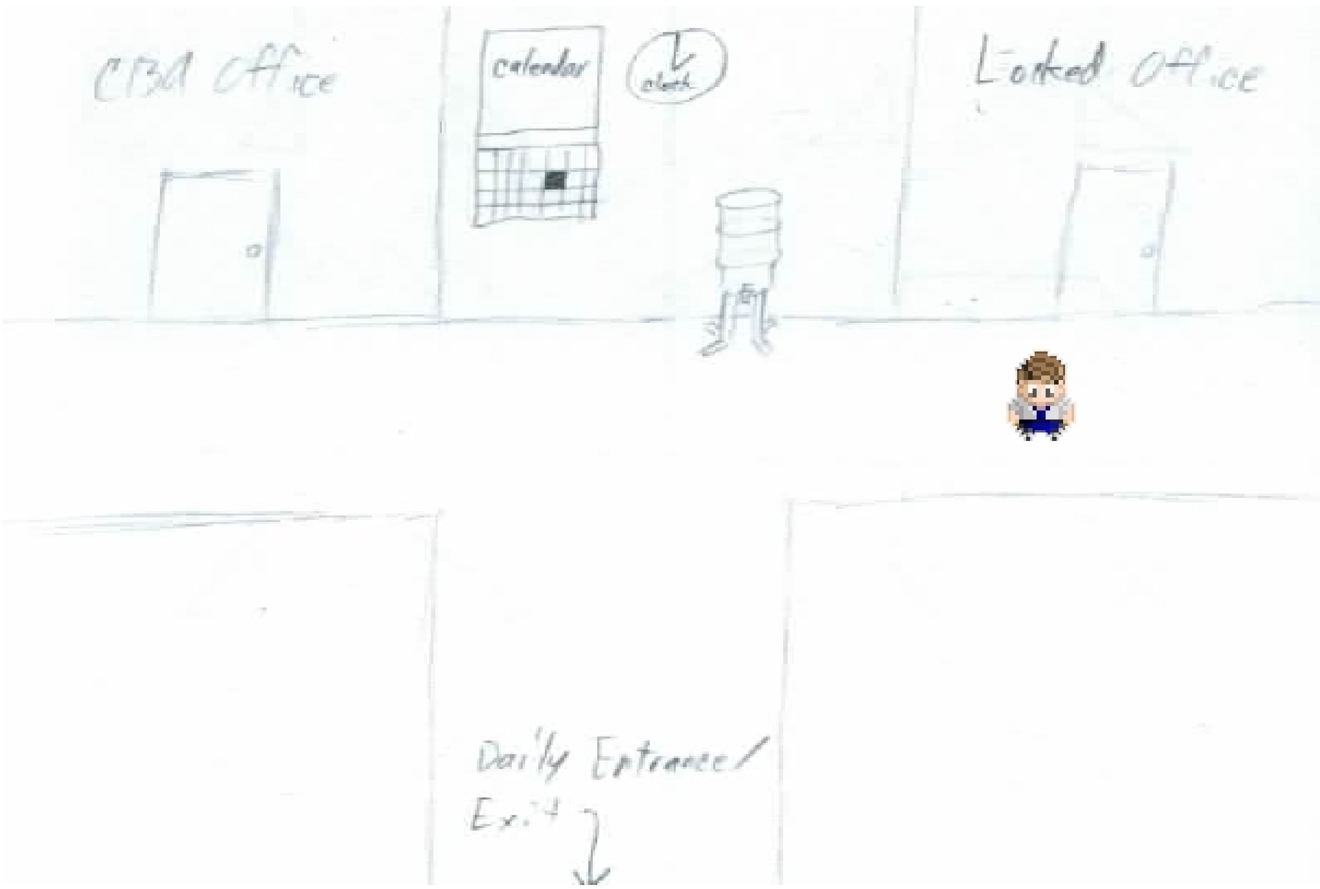


The CBA moves to his office. Movement slide.

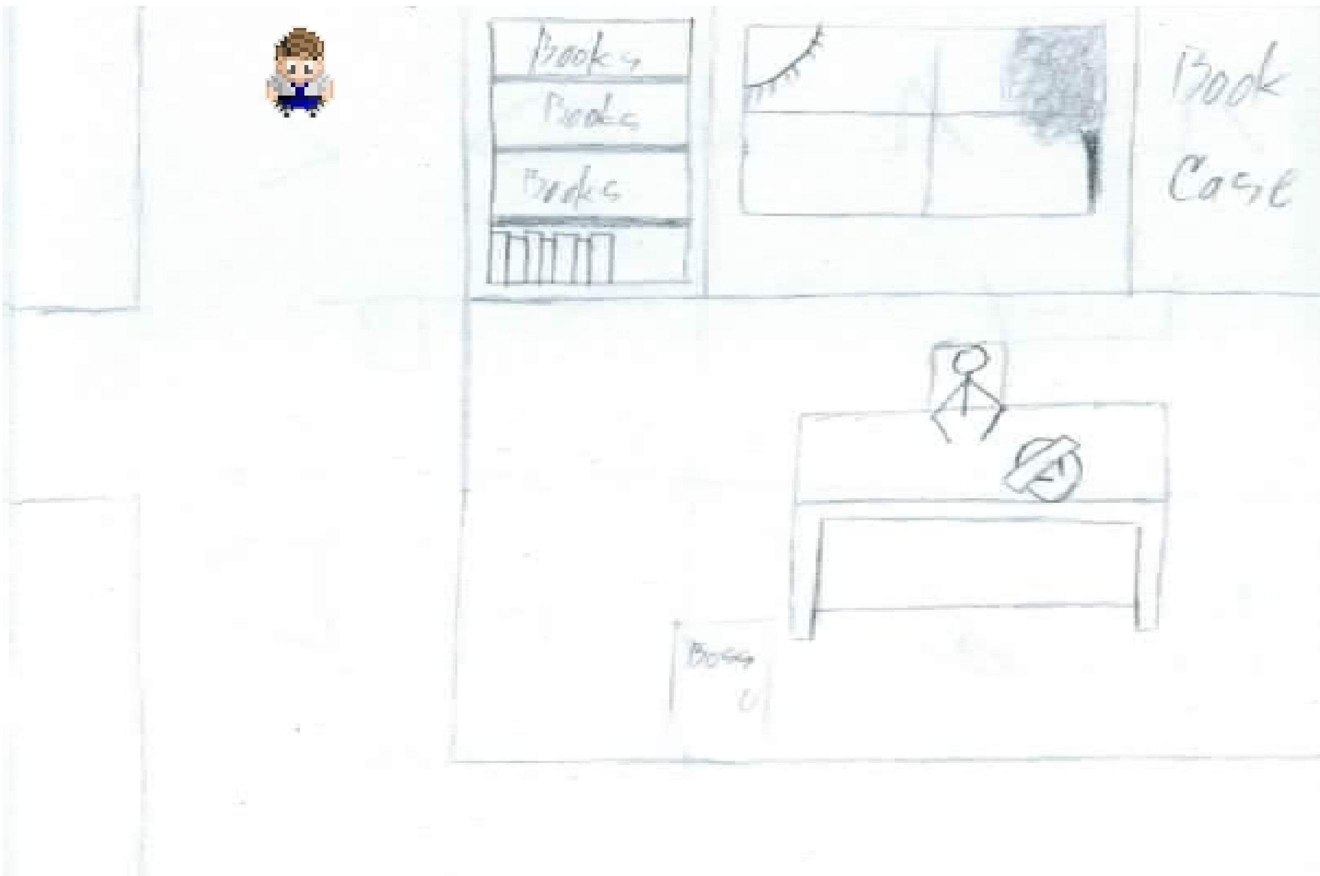
Upon entering his office, the CBA automatically sits at his desk. This is required before starting the day's work, which requires the CBA to travel to the HUB. Movement Slide



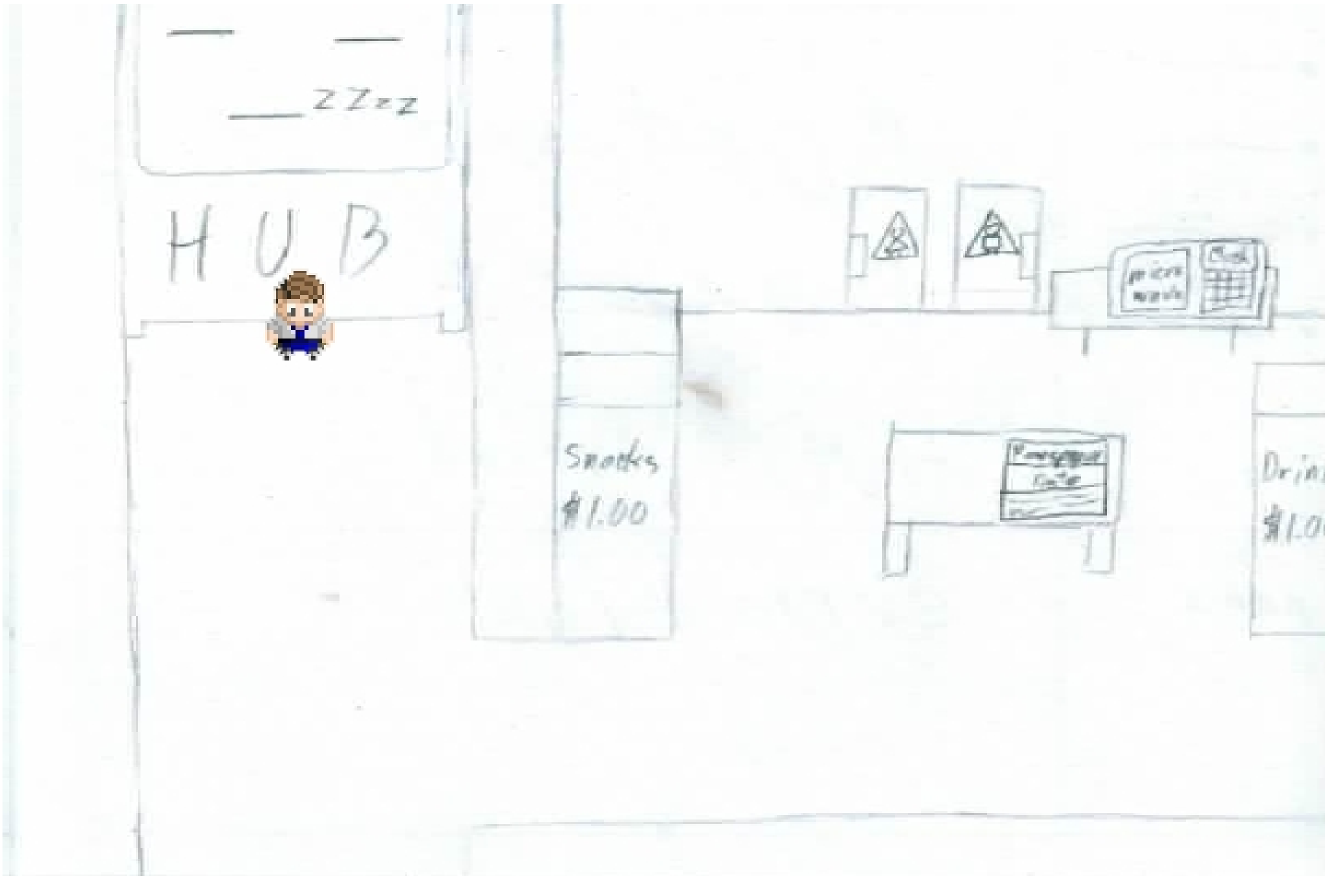
The CBA leaves his office and travels towards the HUB. Movement Slide.



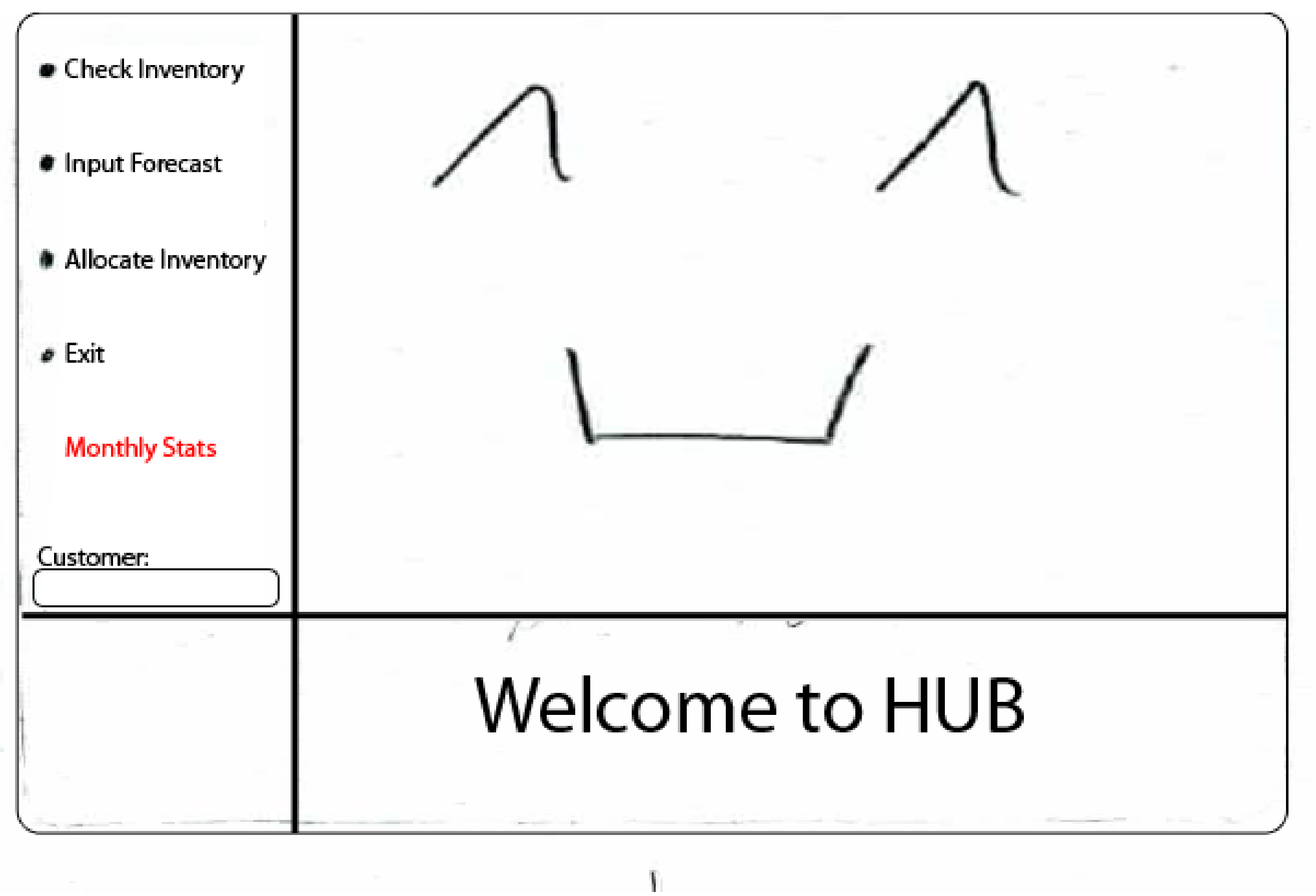
The CBA continues toward the HUB, passing his Boss' office. Movement slide.



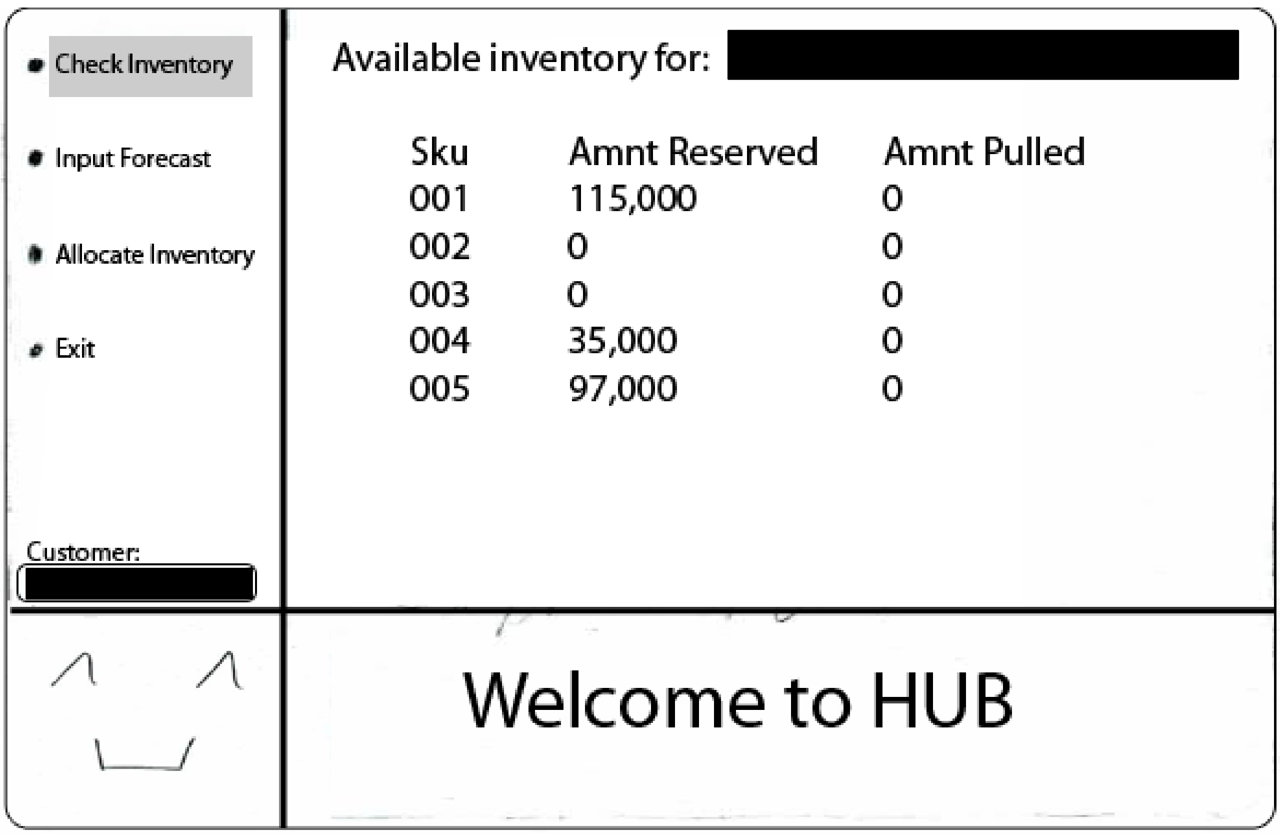
The CBA arrives at the HUB. He approaches it and access it. Movement slide.



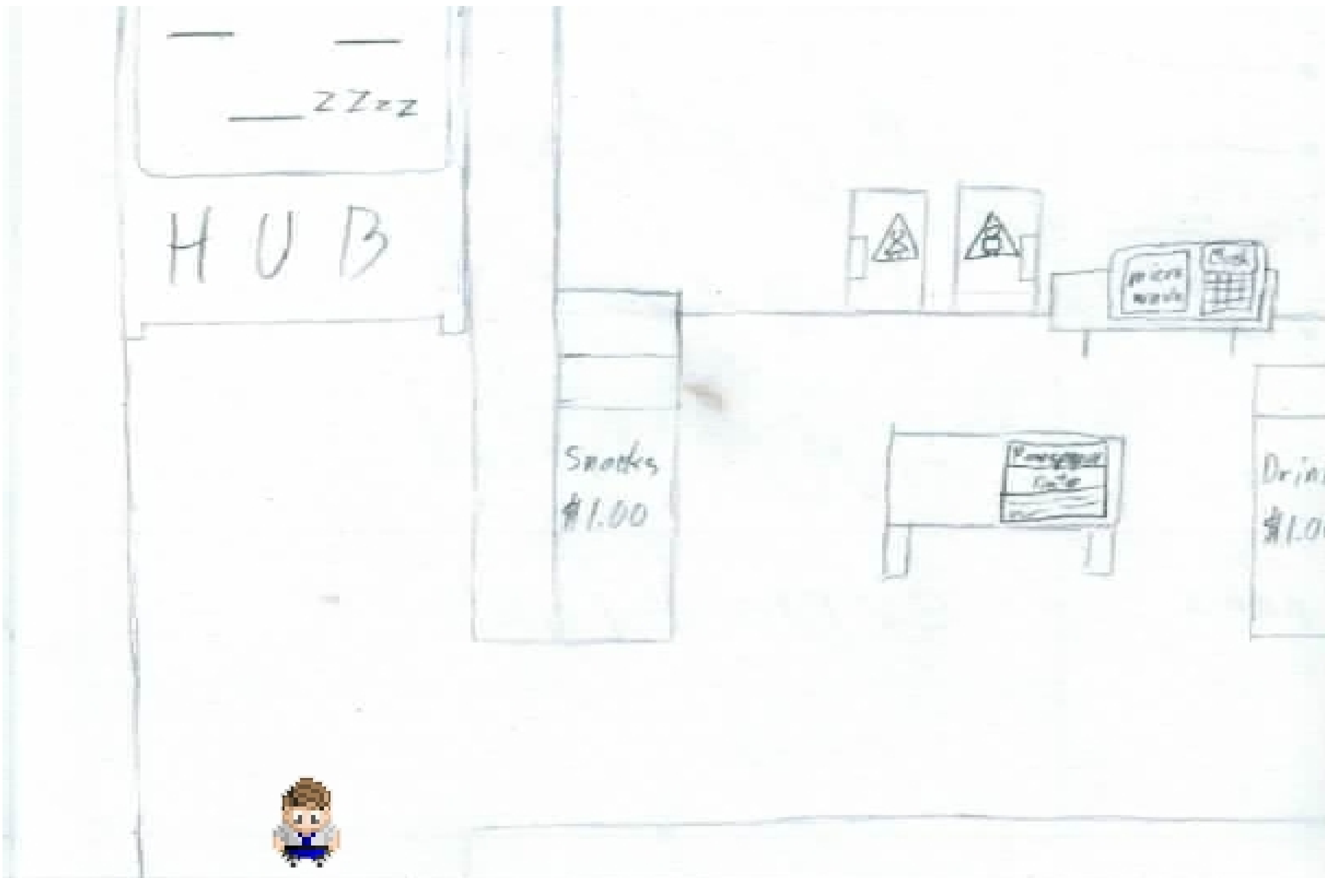
The CBA logs into the HUB, who greets him with a simple geometry-influenced smile. Hopefully, this is reminiscent of the MCP from Tron. This is the first image the CBA will see any time he access the HUB. “Monthly Stats” is unselectable, since it is the beginning of the month. Before the CBA can continue, he must enter a customer name in the “Customer” textbox, then select an option.
This can be complicated by HUB being down, which will be indicated by a “sad”, “angry”, or “sick” face instead of a smile. The CBA will need the Boss' help to fix this.



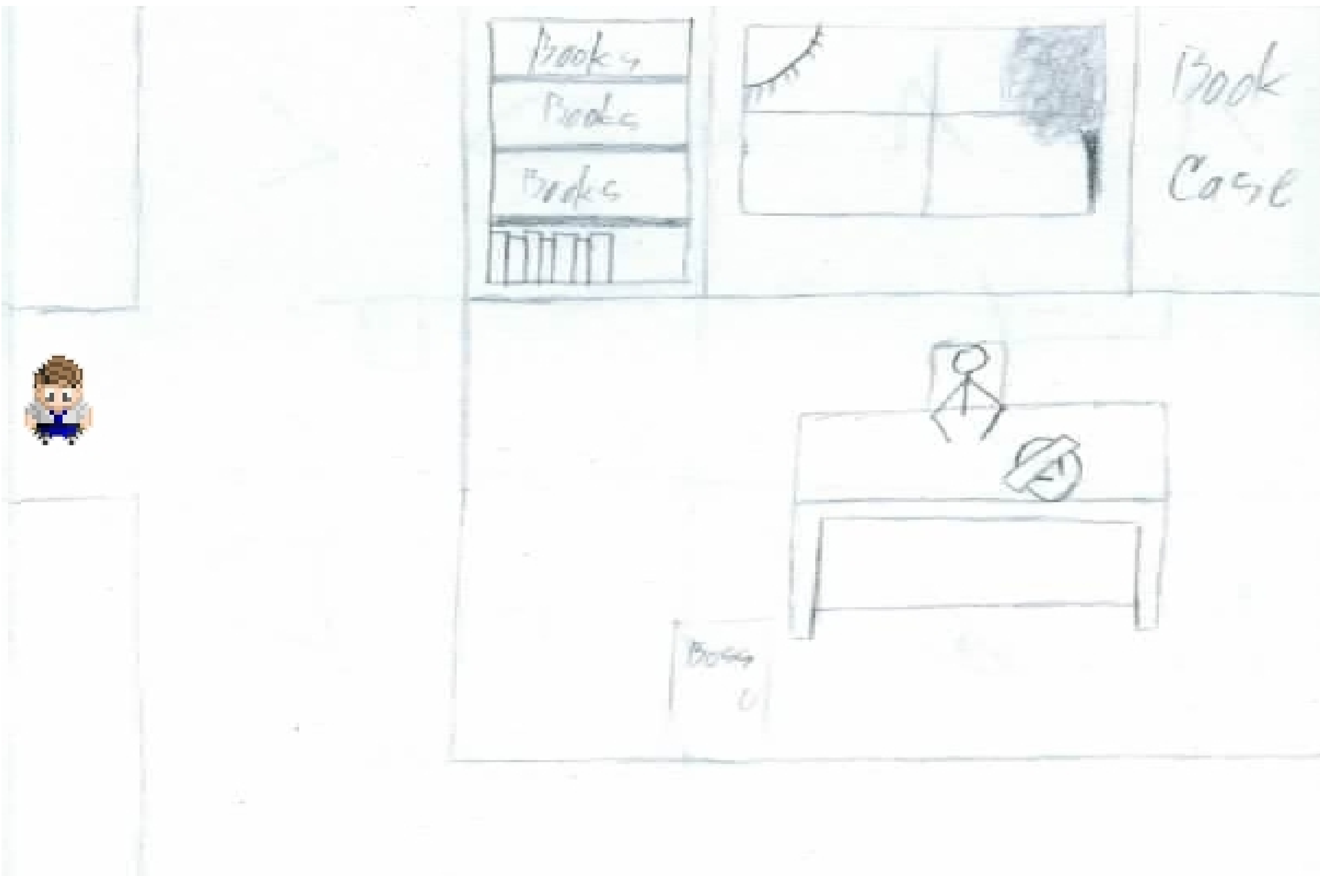
[SafteyStockAllocation]
The CBA checks how many of each SKU is reserved for the given customer. He will use this information later to inform the customer how much is available to pull for that day.



After the CBA is finished with the HUB, he returns to his office. Movement Slide.

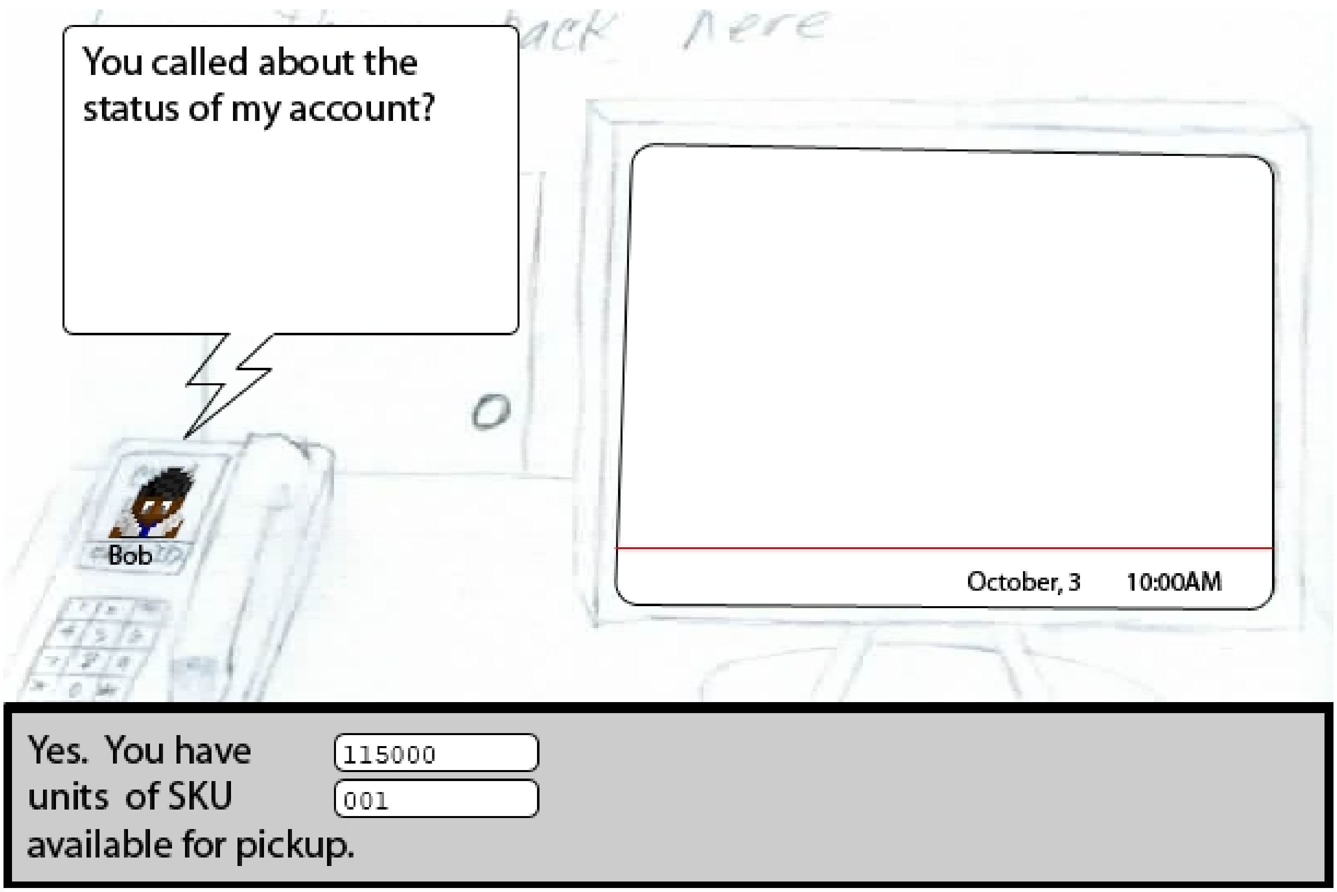


On his way back to his office, the CBA passes his Boss' office again. Movement Slide.

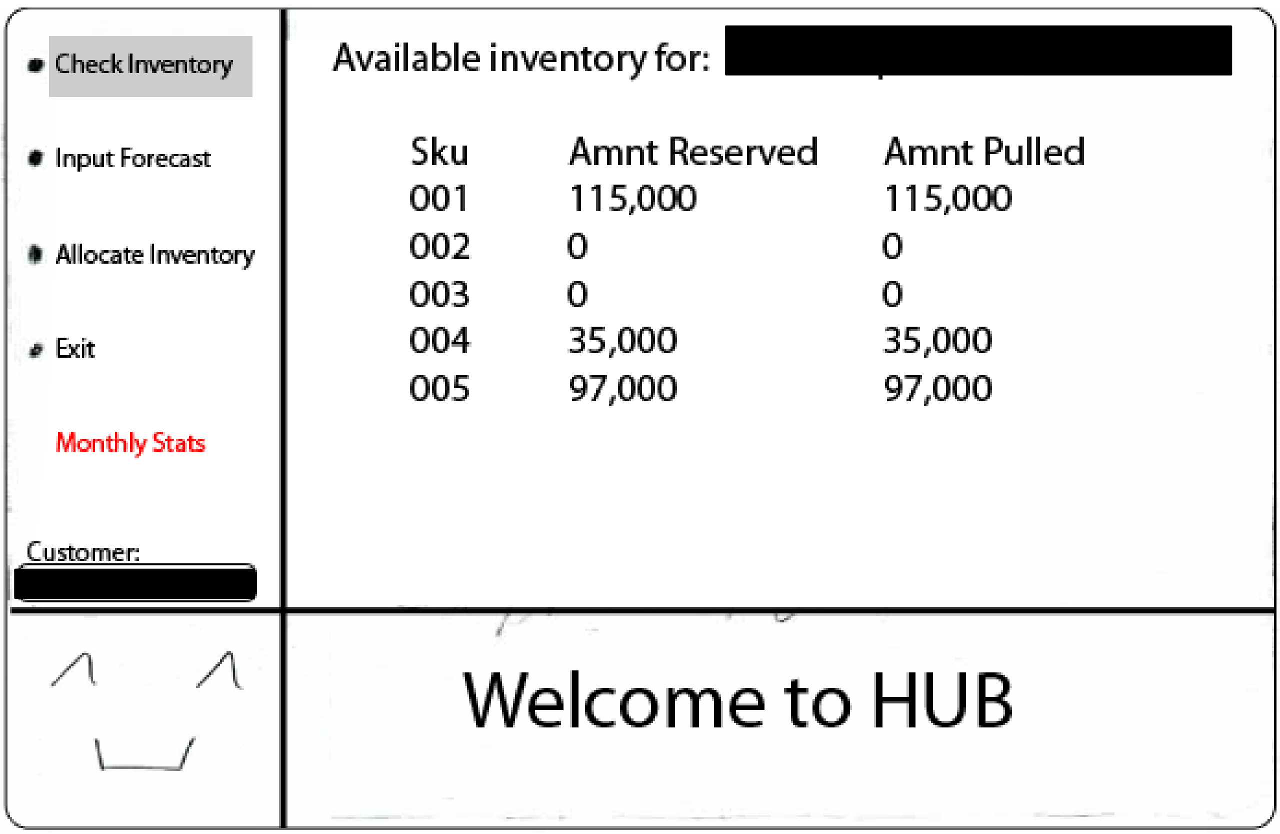


The CBA enters his office again. Movement Slide.

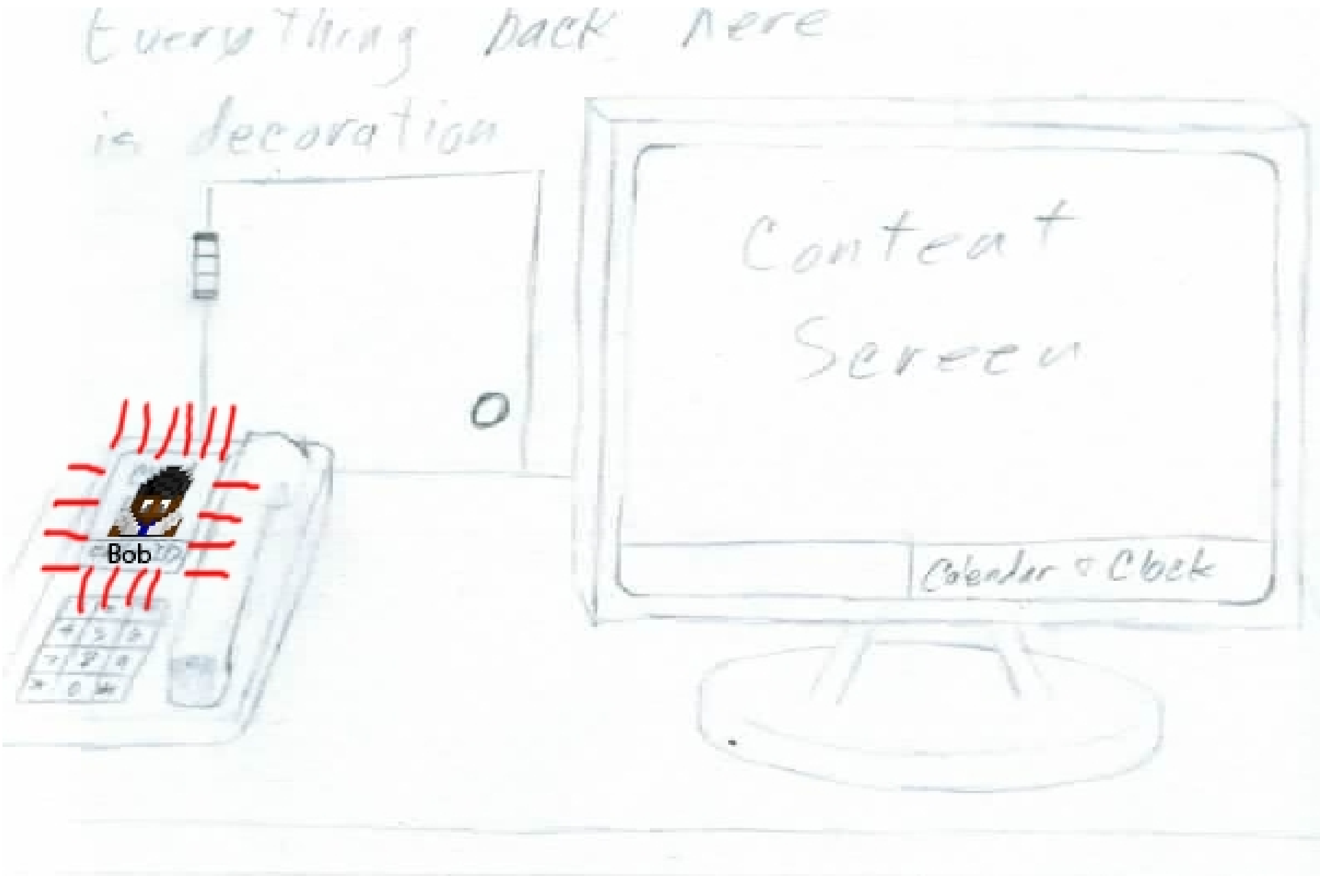
[DailyBOH]
Using the information obtained from the HUB, the CBA calls the customer and tell him how many of each item are available for his company to pull for that day. Giving him incorrect information will result in the boss scolding the CBA.
This can be complicated by the customer not answering or having a bad connection.



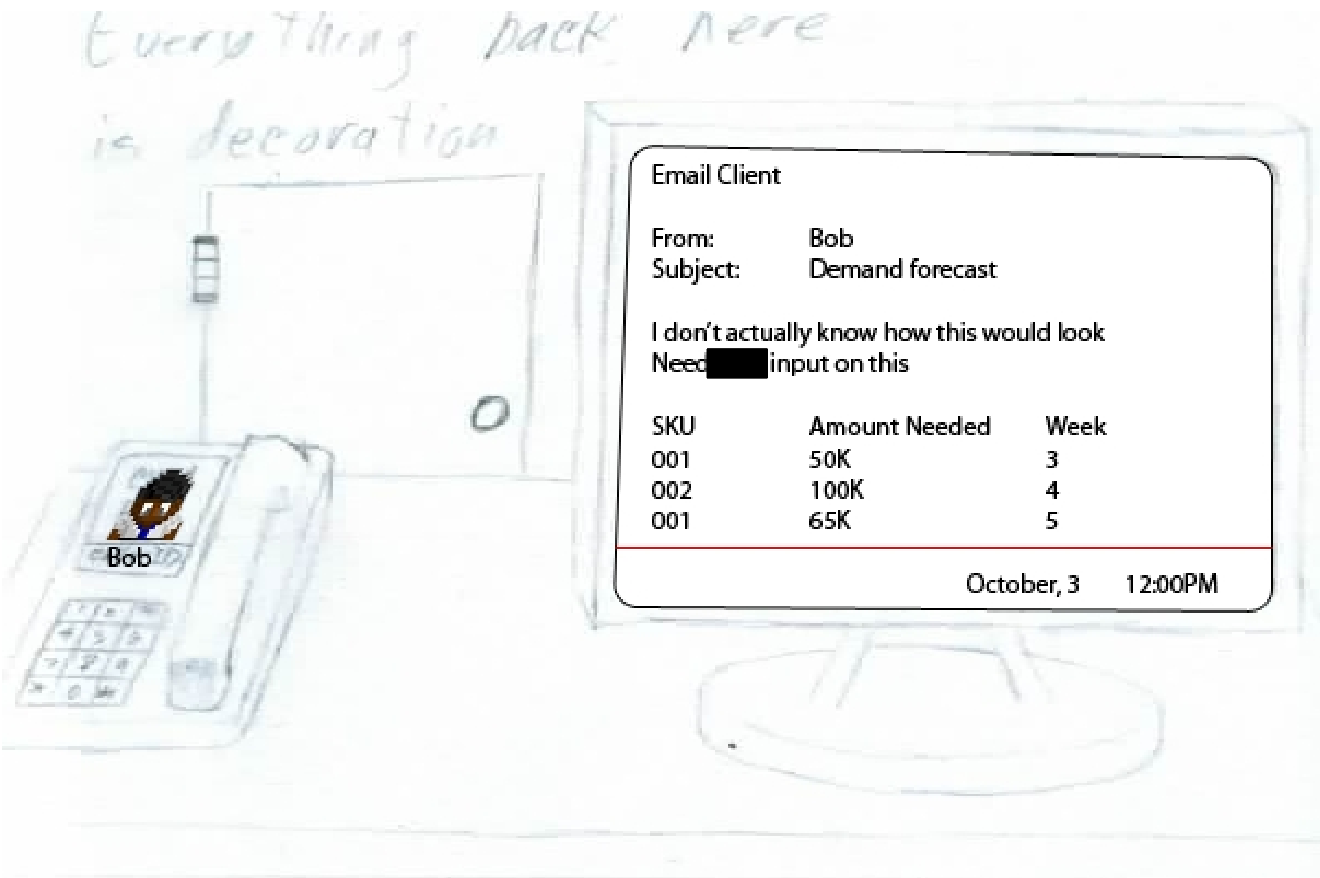
[OrderFulfillment]
The CBA will need to check the HUB to see if the customer has pulled all of his available inventory for the day. Most days this should be 100%, but some days, less will be pulled by the customer. This must be done for each customer before the end of the day.
This can be complicated by database errors.



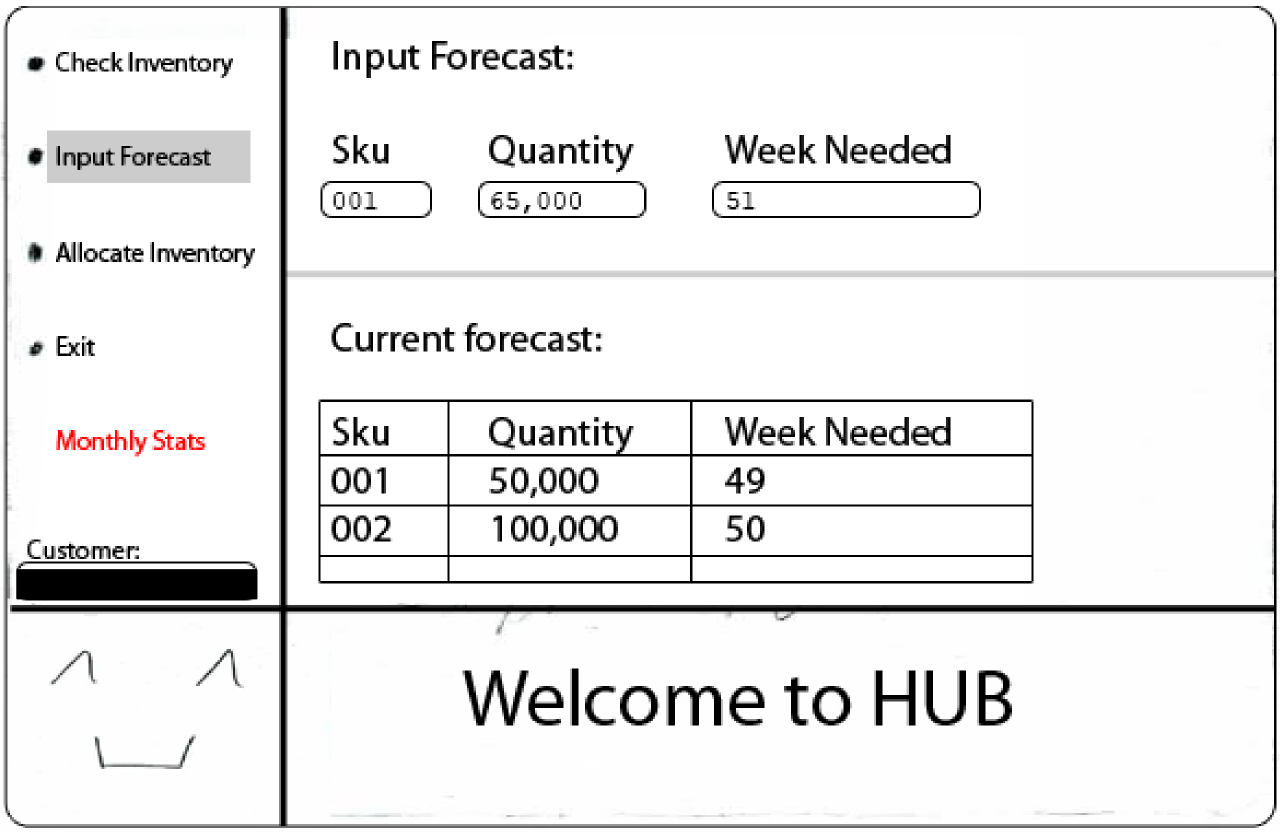
While in his office, a customer calls the CBA. His name, Bob, appears in the CallerID area, and Bob's image appears in the image area. The CBA answers the phone. Bob will say what company he works for.



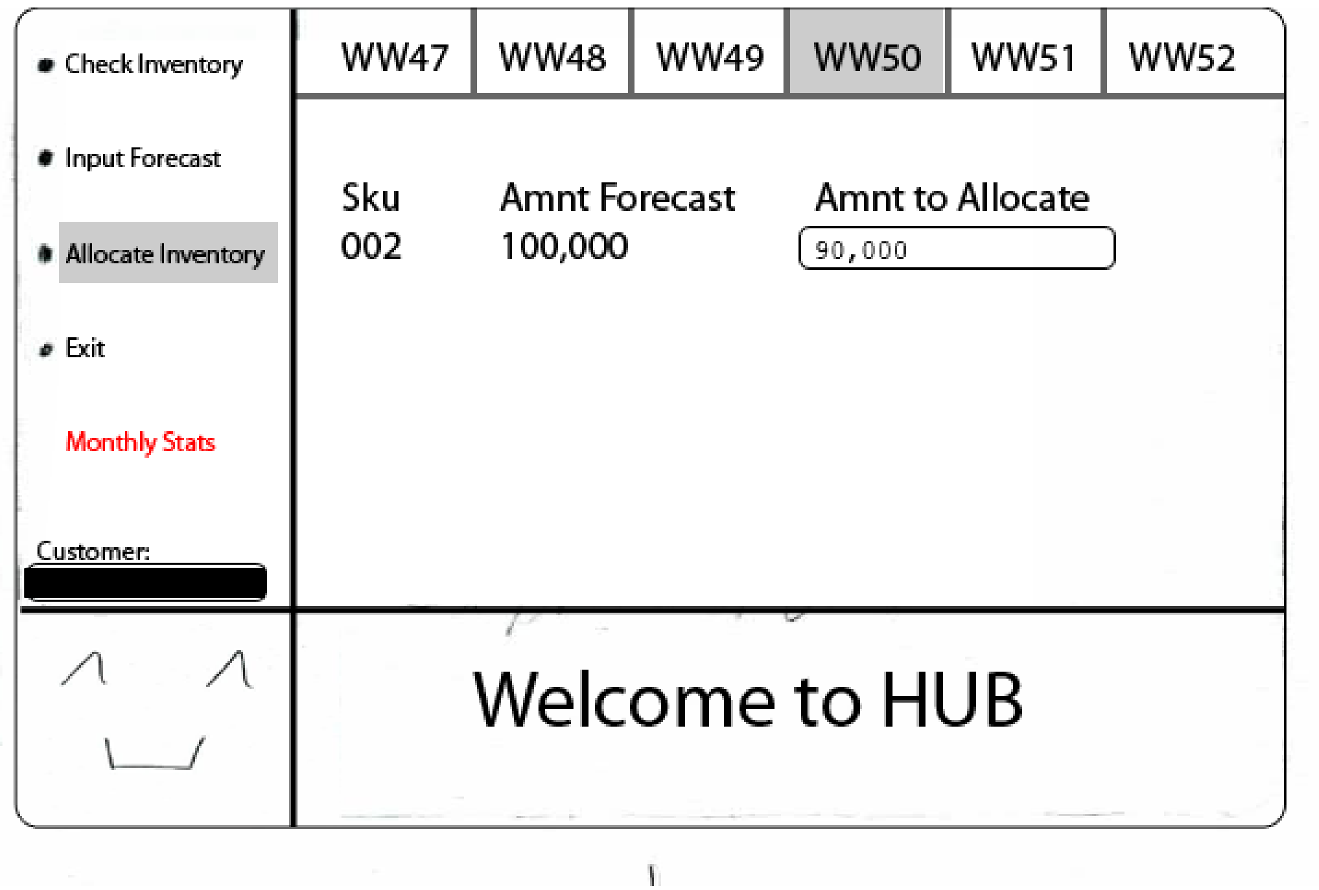
[Demand Forecast – part 1]
Customer Bob emails the CBA his Demand Forecast. It is the CBA's job to remember or write down this information for later use. With the Demand Forecast, Bob must go to the HUB to input the data. This can be complicated by a static-sounding phone connection, or the CBA's computer crashing.



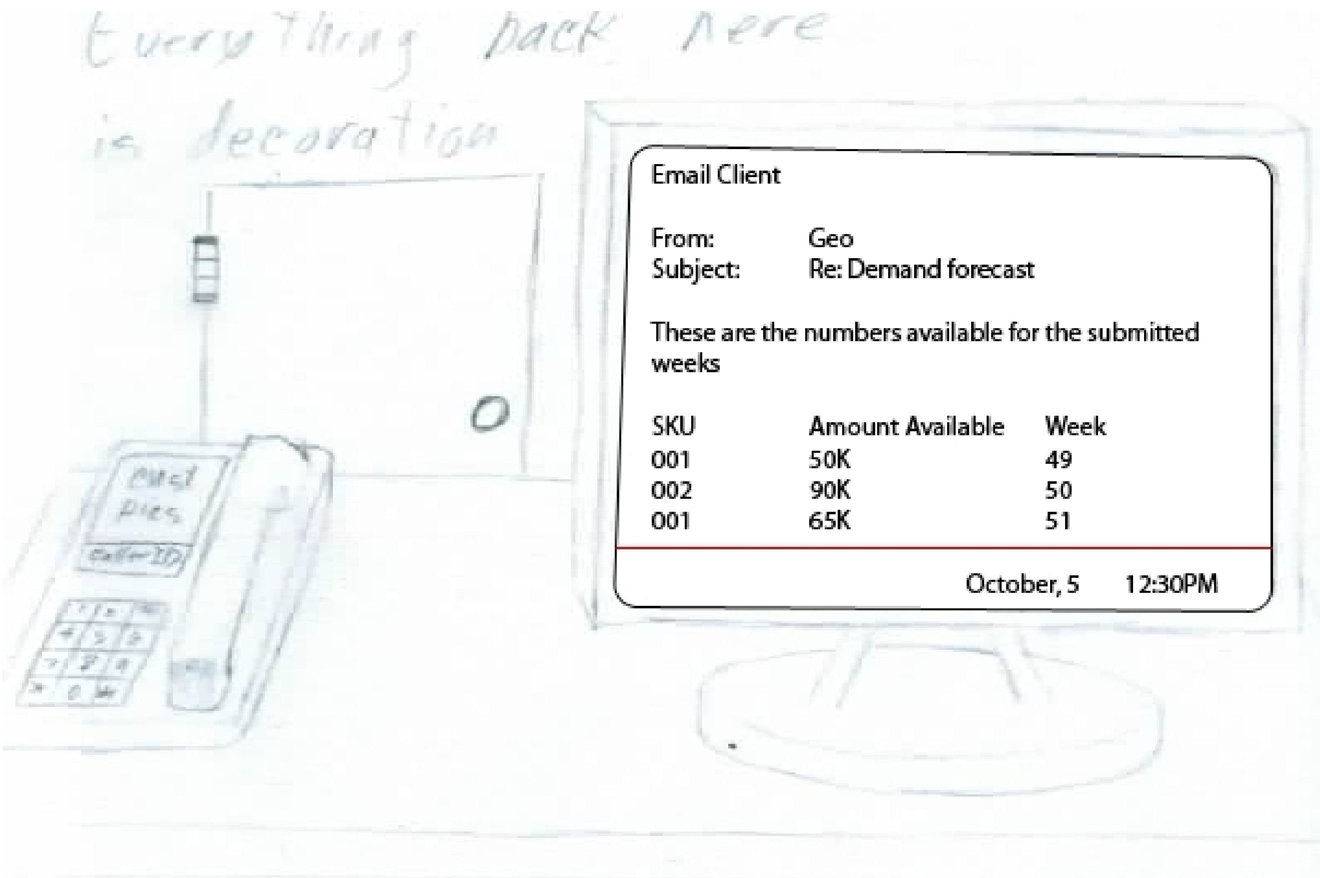
[DemandForecast – part 2]
The CBA inputs the Demand Forecast information from the customer's email. If the input values are invalid, the HUB will make angry faces.



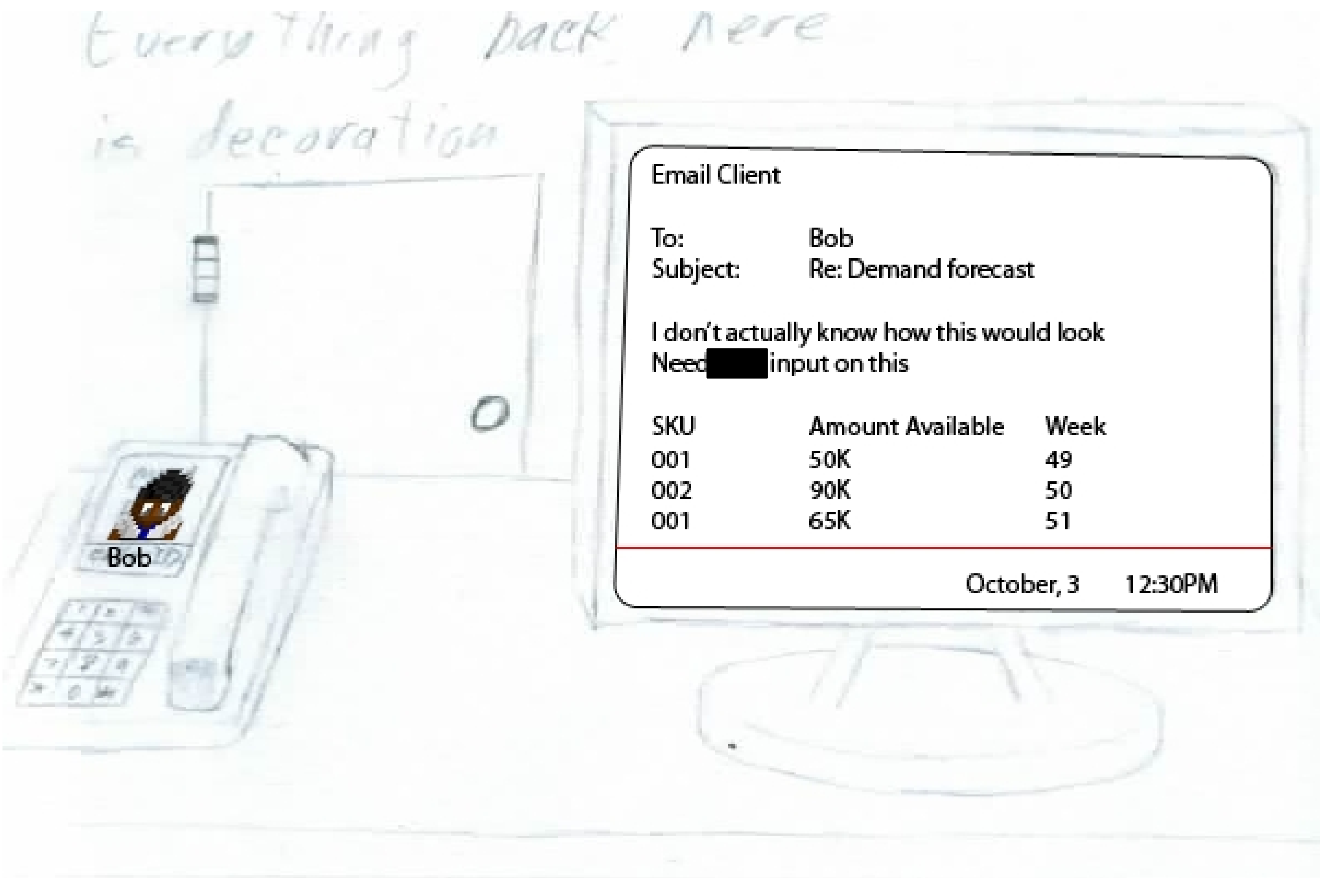
[AllocationEscalation – part 2]
Using the information retrieved from Geo, update the customer's allocated number of items. If the input values are invalid, the HUB will make angry faces.



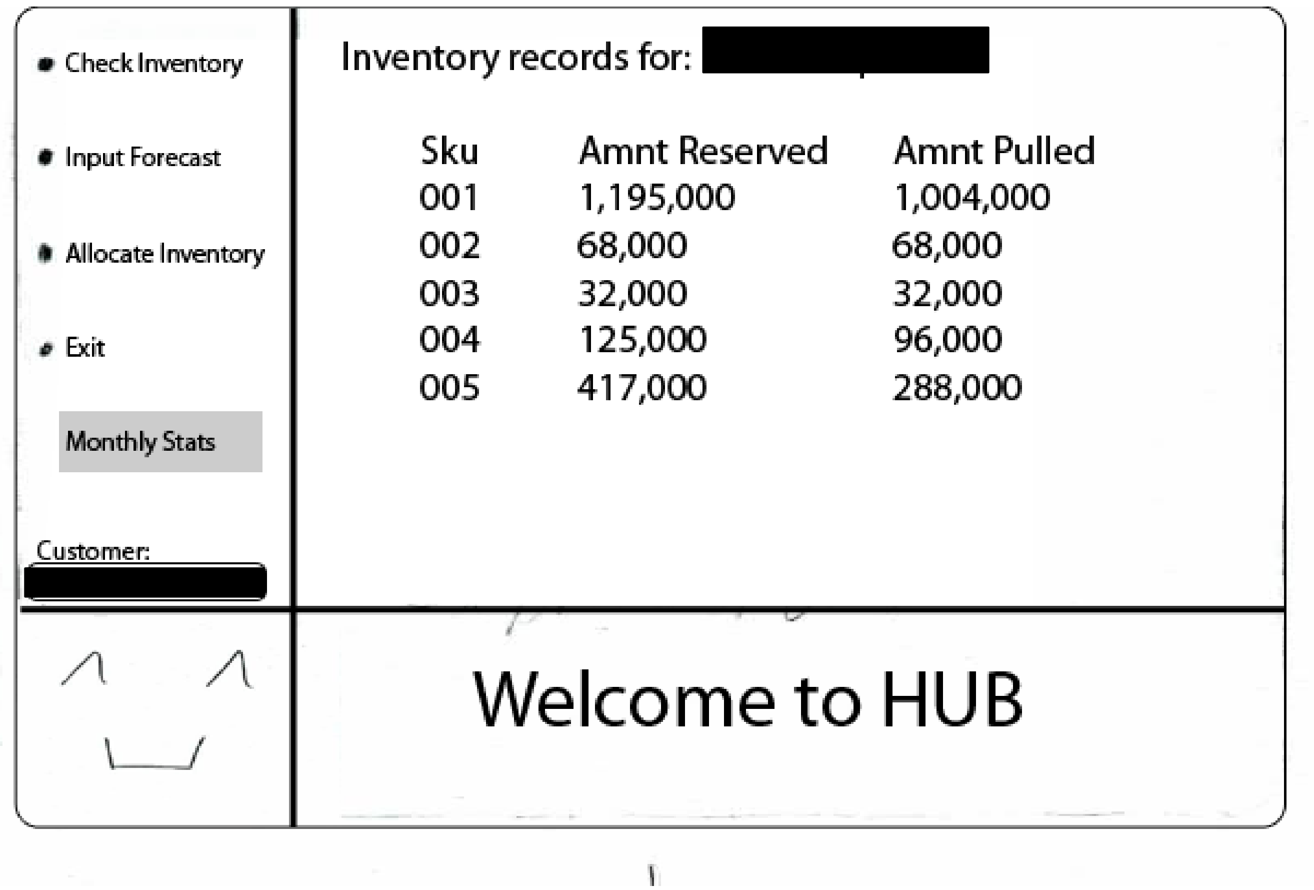
[AllocationEscalation – part 1]
The Geo will send the CBA an email later in the week. This email contains information the CBA needs to input into the HUB pertaining to the customer's allocated number of items.
This can be complicated by the email server being down.



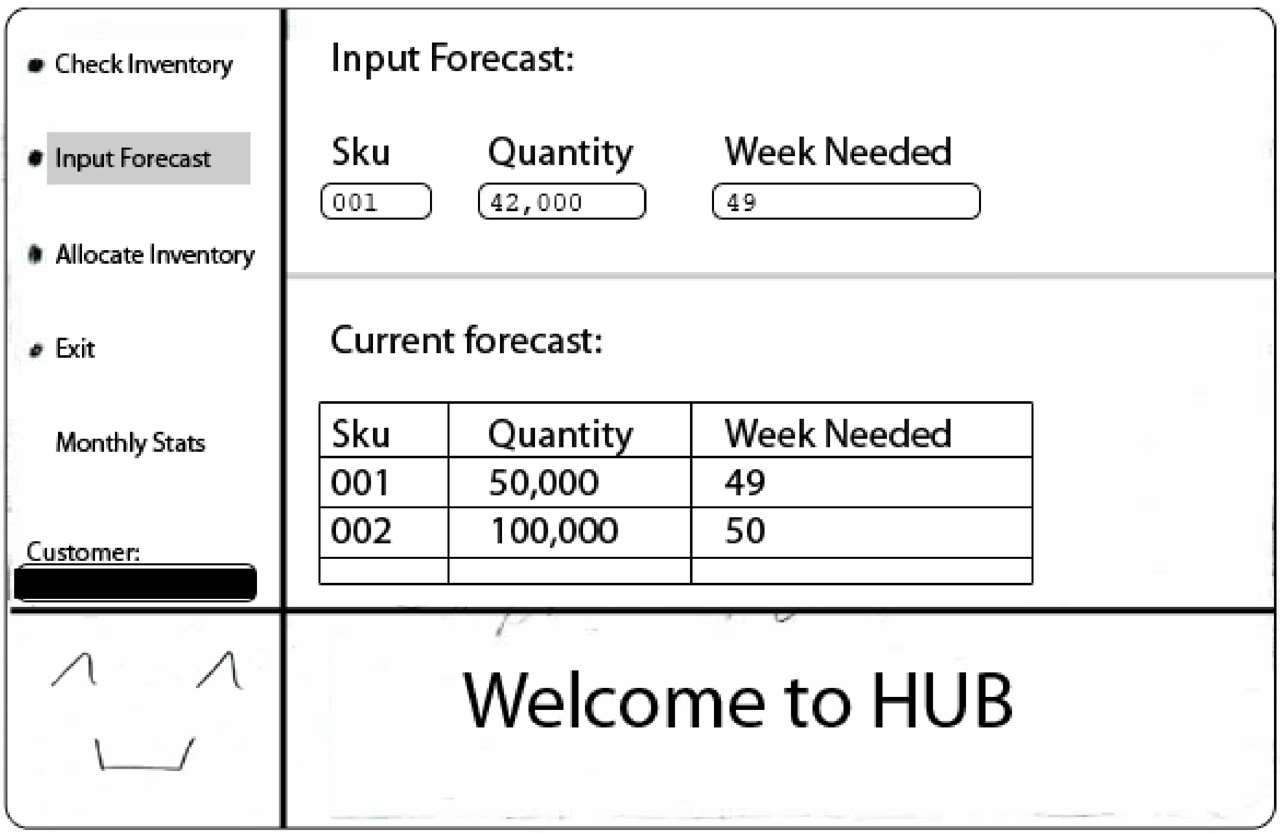
[ResponseToForecast]
After obtaining the Allocation Escalation, the CBA calls his customer and then emails him a Response to Forecast.
This can be complicated by the customer being unavailable, or the email server being down.



[Allocation – part 1]
Starting at the end of the month, the CBA examines the customer's total amount of reserved product versus its actual amount pulled for the previous month. The CBA uses these numbers to know what percentage to modify the customer's Demand Forecast for the next time period.



[Allocation – part 2]
The CBA modifies the customer's Demand Forecast according the percentage that he found earlier. Inputting a number outside of an acceptable range will result in the boss scolding the CBA.



Collaborative Game Environments for Informal Science Education: *DinoQuest* and *DinoQuest Online*

Walt Scacchi and Robert Nideffer
Game Culture and Technology Laboratory
University of California, Irvine
Irvine, CA 92697
{wscacchi, nideffer}@uci.edu

Joe Adams
Discovery Science Center
Santa Ana, CA
2500 N. Main St. Santa Ana, CA
714-542-2823

Abstract: We describe concepts and results that arose from the development and deployment of a large-scale collaborative game environment called *DinoQuest* and *DinoQuest Online*. As an interoperable game environment, DQ and DQO provide a unique experience and approach to informal science education. DQ and DQO are games for helping school-age children to learn about science (or more specifically, life science and dinosaurs). In this paper, we identify and examine different collaborative group forms that emerged to play DQ and DQO, as well as the affordances that help facilitate collaborative game play. Along the way we provide examples of the collaborative groups, affordances, and game play from DQ and DQO.

Introduction

We have developed a large-scale collaborative game environment deployed in a regional science center that joins physical and online activities in the domain of informal science education. The Discovery Science Center (DSC), located in Santa Ana, CA[1], is a regional science center that families and school groups visit in order to experience a diverse variety of interactive science exhibits. These exhibits bring scientific subjects or concepts to life in a hands-on, fun, and entertaining manner. DSC focuses on interactive exhibits as opposed to passive exhibits of scientific artifacts as might be found in a museum that memorializes the history of scientific concepts, scientists, and inventions. DSC is also situated in municipal region, Orange County, CA, whose population spans large concentrations of ethnic immigrants (from Mexico and Latin America, Asia, Middle East, etc.), as well as very affluent to very poor communities. During 2005, more than 275,000 people engaged in DSC visits or outreach activities, while in 2007 more than 425,000 people were engaged, including 150,000 K-12 students of which nearly 90,000 participated in school group visits to DSC. Thus, DSC [2008] exhibits and educational outreach programs are tailored to meet the interests of different communities, age groups, school educators, and other constituencies.

In 2004, effort began at DSC to develop a new interactive game-based exhibit that would focus on dinosaurs as the basis for introducing, demonstrating, and engaging visitors with the concepts from life science (e.g., skeletal systems, elements, and function; digestive system; prey-predator relationships). The life science concepts selected for presentation in the exhibit were those that correspond to curricular topics found in K-6 grade science education standards for California, which are nearly identical to the National Science Education Standards[2]. The UCI Game Culture and Technology

1See, for example, http://www.answers.com/topic/santa-ana?cat=travel
2 http://www.nap.edu/readingroom/books/nses/overview.html. These standards are not part of recent U.S. Federal initiatives like "No Child Left Behind" nor are they the basis for testing scientific knowledge by school grade. Instead, they focus on

Laboratory [UCGameLab 2008] was invited to join this project at this time, and project went into design and operational planning in early 2005. This exhibit was designed to enable the development and deployment of both a physical game-based interactive exhibit at the DSC that would be linked and integrated with a Web-based online game environment. The physical exhibit called *DinoQuest* (DQ) became operational in mid 2006, while the online game environment called *DinoQuest Online* (DQO), went into full-scale operation in 2007. Both DQ and DQO were conceived, designed, and deployed as collaborative science learning game (SLG) environments, and can be evaluated as such. The remainder of this paper focuses on examining and explaining DQ and DQO as collaborative SLG environments, as well as what facilitates different forms of collaboration and collaborative game play.

Informal Science Education through Science Learning Games

Informal Science Education (ISE) is concerned with providing and experiencing scientific concepts, methods, and devices drawn from different science disciplines in settings outside of school, where formal science education occurs. Science centers, museums, after school clubs, and public media (e.g., the *Nova* television series broadcast in the U.S over the Public Broadcasting System/PBS) are the common settings for ISE, though ISE can also occur at home in settings with family or friends. What is key to ISE is that it is elective, discretionary, and a matter of free choice in terms of the content provider, as opposed to schools whose choices may be determined by school boards or others. However, in our view, science centers that showcase interactive, hands-on exhibits are an ideal setting to deploy SLGs, as part of an overall environment for ISE that is readily accessible to a large public audience.

SLGs are games first and foremost. They are conceived, designed, played, and analyzed as computer games, rather than educational courseware, simulations, or interactive presentations of science concepts [cf. WDIL 2008]. As such, criteria such as intrinsic motivation, effective game play balance, and fun [Malone and Lepper 1987, Koster 2004, Salen and Zimmerman 2003] were among our requirements for DQ and DQO, as were other criteria on computer games as art, culture, and open source development practices that we have been investigating [e.g., LaFarge and Nideffer, 2002, Nideffer 2002, Scacchi 2004].

SLGs are a small and mostly marginalized genre of computer games when one looks at the international computer game industry. No companies appear to be making millions of dollars from their best-selling SLGs. In fact, most of the large, well-known computer game companies avoid developing games that are envisioned as "educational" and targeted to specific age-skill groups. Instead, they more often seek to develop games that are fun, entertaining, and engaging, as well as focused on fantasy worlds, rather than on education and academic subjects. Subsequently, there is comparatively little industry interest in developing and deploying educational games in general, and SLGs in particular. However, as some game scholars and educational theorists have observed, many computer games succeed because they are great learning environments that embody both classic and modern theories of constructivist learning, self-identity through role play, reflective thinking, domain-specific specialist language skills, and multi-player socialization [Bainbridge 2007, Gee 2003, Shaffer 2006]. Thus, we see the emergence of the so-called "serious games" community of small, independent game

identifying for teachers, parents, and others what scientific concepts and practices students should be taught and learned (hopefully) in order to become scientifically literate citizens through their K-12 education. Students who excel or become enthusiastic learners of such materials may then be prepared for college level study and a career in a science, technology, engineering, or mathematical field.

development studios and academic/industrial research groups that are beginning to invest resources into development of SLGs. But serious games need not be fun [Bogost 2007] nor collaborative to be effective. In contrast, our choice was to develop and deploy SLGs that would be both fun, collaborative, and oriented to free-choice science learning, which is generally a core requirement for interactive exhibits deployed in a regional science center [cf. Allison-Bunnell and Schaller 2005].

Collaboration through Games and Gameplay

Many games focus on single players, while others seek to encourage multi-player game play. Learning and educational experiences are also similarly focused on individual students or small groups. But where and how does collaboration or collaborative game play fit into such engagements?

Massively multi-player online games (MMOGs) are widely recognized for fostering modes of collaborative multi-player game play as one of their core features and modalities of experience. However, collaboration, in the form of multi-player online discourse and social interaction, often happens around the game to enable collaboration within the game [Nardi and Harris 2006]. Thus, reading, communicating, and interacting with others through multiple media/modalities [cf. Rieber 2005] become a capability for enabling distributed cognition and learning, both through and around game play, whether online or offline. Based on our observation and development of interactive exhibits at science centers, museums, and other settings, we have found the following kinds of collaborative groups are commonplace:

Collaborative group forms

Family group – a family is a problem-solving group that can involve members of different age groups that span generations (children, siblings, parents, grandparents, relatives, and others). Family groups may act in a hierarchical manner when some members (e.g., parents who can read) direct the activities of other members (children who cannot yet read) based on prior family arrangements or skill.

School classroom group – a school classroom group may engage in collective problem-solving when teachers or teaching assistants direct the attention of a classroom group (students) to focus on individual, collective, or competitive problem-solving tasks in order to reveal their comprehension (or lack thereof) of the problem at hand, and to propose a solution to it.

Inter-family group – a multi-family or inter-family group involves members from two or more family units that typically do not have a prior hierarchical relation between them, so that any joint action between families must be negotiated in some manner in order to proceed and benefit one or more of the participating families.

Mixed reality group – an mixed reality group consists of people in a physical setting who interact with other people seen or engaged online through their multi-media avatars (online representations or "ghosts" of the individual people participating/rendered online only). The avatars may be under live, real-time control by remote persons, or may be pre-recorded audio-video sequences controlled through online agents whose actions or behaviors are scripted in advance to respond to user input events.

Online multi-player group – an online multi-player group arises when all persons participating in a group act individually or collectively through online characters, and these characters may shown with human-like or animal-like appearances.

Project development group – the group of people from often different organizations who are brought together for the purpose of conceiving, designing, producing, deploying, operating, and sustaining an interactive exhibit or shared computing application.

However, the emergence of these different forms of collaborative game play groups is not simply spontaneous. Instead, their emergence is facilitated by a number of affordances that not only situate a hands-on interactive science exhibit or SLG, but help make such collaboration fun, engaging, playful, and entertaining.

Affordances for game-based collaboration in groups

Venue—part of what enables small groups to interact, communicate, and collaborate is the venue or place where they meet. Places like a science center convey a set of expectations to many visitors about what this place is about (e.g., observing, touching, or interacting with hands-on exhibits, and being amused along the way), and what kinds of individual or group activities are accommodated (talking, adults guiding the activities of children with them, seeing other people interacting with other exhibits to help learn how to interact or experience an exhibit, etc.). Science centers are places people freely choose to visit, typically as a family unit.[3]

Game genre—a collaborative game is generally a multi-player game or game genre (e.g., adventure/quest game, role playing game, first-person shooter/action game) that can be easily recognized by potential players who may already "know" how to play or what to do as a team, and thus can quickly learn how to start playing the game at hand.

Game exhibit and content—a game exhibit entails both the means and ways for engaging and experiencing the game while going through the game's content. The means are the form of the exhibit —its devices, apparatus, mechanisms, etc.—while the ways are navigational or interactional cues provided to players in order to move through the game's work/play flow.

Game infrastructure—behind the scenes of an interactive game-based exhibit is some arrangement of information technology (hardware and software) and telecommunications (networking) systems that enables the ongoing operation or use of the game by large numbers of players over different periods of time.

Game play situation and participants—each game provides a context and experience for its players, who may differ in their skill, mastery, or interest in the game. Part of the context and experience is conveyed through the game play activities and tasks that players are expected to perform in order to advance or make progress towards completion of the game (at least in games that are goal-directed).

Thus, another requirement for developing and deploying SLGs at a science center is to support such a variety of collaborative group forms and affordances. Now we turn to provide examples of the collaborative groups and affordances that emerged through DQ and DQO.

DinoQuest and DinoQuest Online

Collaborative groups, affordances, and game play for DQ and DQO can be seen through a number of examples. The examples that follow are organized and presented first for DQ, second for DQO, and last

3 DSC does not admit unsupervised children under the age of 16. Thus children come with their parents, older siblings, other family members, responsible adults, or school classroom groups.

for DQ and DQO jointly. Similarly, these examples help document where and under what conditions collaborative game play can be easily observed by outsiders, and engaged by insiders (DQ and DQO game players).

DinoQuest (DQ)

The *DinoQuest* venue – The DQ venue occupies more than 30,000 square feet of outdoor exhibit space at the DSC. It cost more than $6M and two years to develop. There are 13 life size dinosaur models, including a partially articulated Argentinosaurus (one of the largest known dinosaurs at over 120' in length), T-Rex (a complete skeletal replica of "Stan BHI 3033" at the Black Hills Institute for Geologic Research), triceratops, smaller raptors, and so on, all based on paleontology discoveries in the late 20th Century. The Argentinosaurus was designed around a bridge so that visitors can walk through it and physically interact with simplified renditions of some of its internal organs[4]. DQ is partially visible by automobiles driving by the DSC, including those of the adjacent Interstate 5 freeway. DSC also routinely advertises its featured exhibits in the print and electronic news media, which all helps to develop awareness, interest, and attendance to DSC and its featured exhibits, like DQ. Thousands of visitors and dozens of school groups each month go through the DQ exhibit as part of their visit to DSC, so that visitors typically go through DQ while other people are doing the same, thus visually perceiving the shared commonality of a DQ walkthrough or play experience.

Figure 1. Partial view of DinoQuest with the Argentinosaurus in view

Quest-style (treasure hunt) game design – DQ is modeled as a quest-style game common to many role-playing games. In a quest, players are given assignments or "missions" to complete, which on completion may earn them rewards of some kind. Given the target audience for DQ is children in K-6 grades, some of these missions are similar in spirit to a "treasure hunt", which then helps set the stage for more advanced missions. In DQ, there are eight missions, organized first into a set of six discovery missions (e.g., locating specified skeletal bones in different dinosaurs), and then two additional more advanced missions. The missions are selected at the "Field Station," which represents an on-site facility where other quest related materials can be found (e.g., handouts that further explain the learning

4 As there are no known fossilized remains of the internal organs of nearly all dinosaurs, the mechanical characitures are provided as suggestive analogs to help convey that dinosaurs were living creatures with internal organs whose overall purpose may be similar to that found in humans.

objectives of each mission). See Figure 2.

Embedded sensor network and sensor activators – DQ utilizes an embedded sensor network with more than 120 embedded sensor sites that are activated when illuminated with an infra-red designator (end-user wand). The design, installation, and programming of this technology for DQ was provided by Creative Kingdoms Inc.[5] All designator wands have unique digital identifiers that are bound to aDQ player at the Field Station before first entering the DinoQuest venue. Sensor illumination can then be reconciled with respect to which end-user wand has illuminated it, along with the end-user's current mission/quest. This enables the DQ environment to determine whether the user has designated, found, or otherwise discovered the appropriate life science construct (e.g., eye socket) or paleontological element (triceratops head). Figure 3 displays the IR designator wand for playing DQ, as well as a sample view of two children using their wand during a discovery mission.

Figure 2. Parent-child team selecting a DQ mission at the Field Station

Figure 3. The DQ Infra-Red designator wand, and its use in a DQ discovery mission

Situated kiosks – Both selection and progress reports associated with each player's current mission can

5 http://www.creativekingdoms.com/

be queried at multi-media kiosks located either in the Field Station or out in the "field." Both selection and progress reporting are triggered by pointing/gesturing the IR wand in the direction of a kiosk. Kiosks in the field are further assigned as by their association to a virtual collaboratory corresponding to each mission and its geographic location (Africa, Korea, Argentina, United States, etc.). In this regard, we have adopted and integrated the recently developed construct of the research collaboratory [Collabs 2008, Teasley and Wolonsky 2001] into a form that is both (a) accessible to both children, parents, and others in the public, and (b) provides them an introduction, awareness, and simulated experience with such facilities.

Figure 4. DQ players interacting with kiosks in the field during their current mission

Interactive multi-media presentations on demand—Each kiosk contains a networked personal computer that contains a local cache of pre-recorded multi-media content. These recordings depict people acting in the role of different scientists (e.g., paleontologist, zoologist) in different research settings (paleontology field site, research laboratory) who then interact with DQ players to review or explain features/goals of their missions in progress.

Figure 5. Science role models as DQ collaboratory scientists

These science role models vary by ethnicity, age, and gender, as shown in Figure 5. The recordings do not reveal answers to the missions until mission completion, but they do provide hints if the player is having trouble completing the mission. As the sensor network monitors each player's discoveries (and misses) and tracks all sensed designation events as transactions to a central database, then when a player queries a kiosk, the system can review their progress to determine which media assets to present to acknowledge progress or to provide some assistance.

Supplementary handout materials for reading and review—The DQ Field Station also is stocked with handout materials. These materials, created by the Education department at DSC, are provided primarily for parents, teachers, and other adults that seek to understand both (a) how to play different DQ missions, and (b) how the missions correspond to California Science Education Standards by school grade level. In general, children players do not consult or review these materials, as it seems they are more comfortable just going out into the DQ venue and playing to discover/learn.

Other DQ visitors – DQ usually is often occupied by dozens of visitors at any time. People in groups (families, school groups) can observe one another and develop a sense about what the others may be doing, and whether these others are having fun, are learning, or seem lost/confused. In particular, we found once DQ became operational that people in different, unrelated groups would at times engage in situated conversation with one another to offer help or advice. Said more simply, people in different groups would collaborate with one another when it seemed they could assist others who might benefit from such assistance. Play in DQ fosters these kinds of situated and emergent collaborations.

Scientific modes of inquiry and field research practices – A central component for developing scientific literacy is developing an understanding for how to engage in scientific modes of inquiry and reasoning. Once again, science education standards include this component at a level appropriate for K-6 grade learners. DQ game play supports various modes of inquiry and field research practice, even within discovery missions. First, DQ missions are thematically organized around topics like identifying trace fossils, and anatomical components and configurations. Second, missions are accumulative—recognizing prey-predator relationships revealed through anatomical features (e.g., small body, long legs, and sharp teeth for meat-eating raptors/T. Rex compared to the large body, short legs, toothless beak for plant eating triceratops). Third, learning (i.e., completing the game) requires observing the environment, gathering facts, and recording or reporting them. Fourth, successful missions can produce results that may be useful later (i.e., reusable) in other missions or problems. Fifth, exhaustive search is not a productive strategy in all situations, and sometimes a search that is focused due to knowledge about features, characteristics, or properties the problem at hand can lead to useful results more quickly. Sixth, scientists come from diverse cultures and may work in distant locations, so someone like you could become a scientist if you have the desire and are willing to do the work required. Last, scientific work entails knowing how to (a) read, communicate, and interact with other people, (b) use tools or instruments, and (c) use telecommunications and information technologies to organize and report data, as well as to communicate and collaborate with others at a distance. Of course, how well DQ players can remember, transfer, and apply such matters in other settings remains an open question.

DinoQuest Online

The DinoQuest Online venue—The DQO venue is a publicly available Web site that downloads a Flash-based DQO game engine needed to play the game. The DQO game engine dynamically loads the content associated with each game module. DQO currently supports 13 game modules. Each module is a game, and much like DQ, the modules are partially ordered and game play results/knowledge are accumulative. The DQO game modules do not mirror the DQ environment or play experience. Instead, DQO provides a set of simulated environment, some literal, other strictly conceptual. Figure 6 provides a view of the in-game home for DQO that appears as a multi-media computing laboratory or collaboratory [cf. Collabs 2008, Teasley and Wolensky 2001]. In the figure, the large multi-panel wall display serves as the in-game interface for “connecting” to remote collaboratories in geographic locations that reflect the same choices in DQ. Selecting one of these collaboratory panels transitions the user the associated game modules. The large map display is the “DinoSphere” which is a higher level, multi-player game space that is accessible only after completing the other game modules. The doors to the right take the user to “MyLab”, which is the user's private in-game laboratory office where their research points and other objects collected through both DQ and DQO are kept for later use. The computer screen in the foreground is an interface to an embedded multi-media presentation from an avatar also introduced in DQ. Most of the 13 game modules can be played by a single user, but based on observations at the DSC, children often play DQO with an adult/parent companion who wants to share the game play experience, engage the child player with a discussion about game play, or seek an explanation from the child about what's going on in the game (more often than not, the child needs to explain how the game works to the parent/adult who doesn't usually play computer games).

Figure 6. DinoQuest Online in-game home

Multi-genre game design—As previously indicated, DQO consists of 13 game modules. Collectively, they take a player about 3-5 hours to play to completion. However, individual game modules vary in the duration, exposition of life science concepts, and game genre. For example, DQO game modules includes games drawn from quest, design/simulation, puzzle, and mini-game genres. However, it may also be fair to say that these games can individually or collectively be viewed as “casual games” that can be started, played for a brief period, stopped, and restarted later. However, game scores and research points earned persist across game play sessions, as long as the same user (identifier) is playing.

Multiple in-game user controls—While DQ benefits from the use of an embedded sensor network and infra-red wand sensor designators, DQO utilizes multiple in-game user controls. For example, as (partially) indicated in Figure 7, the user interface to the fossil dig pit game associated with the Argentine collaboratory employs multiple, task-specific cursors (hand, pick, shovel). The hand is used to select other tools, or to pick up and move (to a storage mechanism) a fossil bone structure that has been "dug up" through game play. The pick is used to only break rocks that appear on the dig pit grid. The shovel is only used to dig into a dig pit grid cell, one at a time. The in-game dashboard on the upper right keeps score of the times used versus available to use in order to self-monitor progress and to create a resource scarcity. These in turn help motivate users to carefully choose when and when to use each user controlled in-game tool.

Figure 7. The Fossil Dig Pit game module showing different in-game user controls (hand, pick, shovel) and a dashboard indicating resource utilization (number of possible uses of the pick and shovel).

DQO game environment—DQO represents a contemporary game platform. It is coded in Flash 8, which runs in most commonly available Web browsers (e.g., Internet Explorer, Firefox, Safari). It is accessed from a single Web site (www.DQOnline.org), which in turn downloads the DQO game engine into the user's Web browser, which in turn downloads each game module and its content on user demand. Many families and school groups access and run the DQO game environment on a desktop or laptop computer, as DQO does not require high-end microprocessors, graphics accelerator cards, or the like. Accordingly, the design of the DQO game environment was conceived to enable the largest possible audience of end users or players, including those who may have older, less powerful computers, which includes many under-privileged schools.

Embedded multi-media content—DQO draws on and shares some of the multi-media assets originally created for presentation on the DQ kiosks. Figure 8 shows an example of an embedded video file that is played on command, and provides a brief explanation of the goals and levels found in DQO. The avatar is the same in-game character that plays the same role in DQ. Use of these in-game characters across DQ and DQO creates a sense of continuity in content and play experience, even though DQ is played in an mixed reality environment, while DQO is play in an online environment.

Figure 8. An in-game scene in DQO with an embedded video displaying featuring an in-character also in DQ that introduces and explains DQO's goals and levels.

Embedded tutorials for teachers and parents—It might be surprising to learn that teachers and parents more often than child players want to know how the game operates by reading some prepared materials. Child players on the other hand, often have little/no problem figuring out how to play each DQO game or how to use the in-game user controls, as game play helps to motivate or explain each in a situated way as needed (i.e., children are willing to try something in the game to figure it out, rather than first reading about what to do). Gee [2003] reports that children who learn to play games in such a manner often acquire deep knowledge of the in-game specialist language, terminology, and game play moves that are difficult to determine by a competent adult just by reading a game manual. However, in order to help satisfy the requests from parents, teachers, and other educators, we added a series of embedded tutorials and in-context explanations to help teachers and parents better understand what their children may already know. Figure 9 provides a display of in-game help that is part of such a tutorial for one of the game modules. Beyond this, as DQO players progress from game module to module, DQO also displays interstitial (and stylized) text panels that provide further contextual information about some of the underlying scientific concepts or discoveries that are recreated in the game. These interstitial images (or cut scenes) also serve to occupy the player group (e.g., child and parent) with a simple diversion while the next game module in being downloaded and readied for play.

Figure 9. An in-game view of a teacher/parent tutorial explaining the goal and process for playing this DQO game module (reconstructing fossilized skeletal bones collected in the Fossil dig pit module)

Contemporary game play practices used to elucidate life science concepts—SLGs, as games, need to be more than just interactive presentation of scientific concepts, or simulations of scientific practices or processes. As such, we sought to find way to utilize both original and familiar game play mechanisms and play practices in developing each of the DQO game modules. In addition to countless hours we have individually spent playing dozens of games of all sorts, we have found some DQO players can readily recognize game play mechanisms and play practices that we adopted and adapted from other popular games. For example, in Figure 10, we see a view of the ecological relationships game module within DQO that enables play with prey-predator and food chain relationships. This module utilizes a “Tetris” style of game play, where a configuration of ecological elements (carnivores/predators, herbivores/prey, and plants) can be rotated as they move from left to right to match up with configurations that have already been anchored, in order to maximize the matches (e.g., carnivores prey on herbivores, herbivores prey on plants, unmatched carnivores die and help nuture plants). As such, we (and many adult players familiar with Tetris games) find this game is both familiar to play, yet at the same time, presents basic life science concepts by repurposing contemporary game mechanisms and practices.

Figure 10. A Tetris-like game for matching ecological relationships like prey-predator and food chains

Multi-person game play—As we have indicated above, it is possible for an individual to play DQO without others. However, it's less fun that way, though it can be quite absorbing and attention grapping for some child players. Instead, when parents/adults want to know what their young children are doing when using a computer accessing information on the Web, they will often join with the child to either play with, share, or engage the player while they play DQO. Beyond this, DQO also features a final level game module, DinoSphere, which is a multi-player or multi-character game module. In this module, player specify and configure a dinosaur of their choosing, using the resources and points they have earned from previous game play. DinoSphere features four ecological niches that serve as simulated physical world environments where different dinosaurs must survive or co-habitate. As players by this point have already learned about life science concepts like prey-predator relationships, then the quickly realize small predators (e.g., raptors) individually are not a threat to larger prey (stegosaurus), unless they can find other similar predators who can then collectively act to surround and overwhelm a larger prey. Figure 11 provides a view of a forest ecological niche within DinoSphere where one small raptor seeks to engage a larger stegosaurus as prey, but without success.

DQ and DQO together

Emerging DQ game play experiences that bridge physical and online activity – many visitors to the DSC and DQ come to find that they can continue their (science learning) game play experience online after visiting DQ. On exit from the DQ venue, pathways guide visitors to walk by a group of PCs all configured to run DQO. Visual signs provide encouragement to sit down and try out DQO on the spot, while other guidance describes how to continue to play DQO at home over the Web at http://www.DQOnline.org. Frequently, DQ visitors who have already gone through DQ grab a seat in front of an available PC and start to play and explore DQO. Other DQ visitors in turn observe the DQO

Figure 11. A scene from the DQO module, DinoSphere, where multiple players or in-game characters (e.g., stegosaurus and raptor) can interact in a simulated ecological niche to survive or thrive. The dashboard at the top displays the status of various resources controlled by the player Tom.

game play in progress, and this often helps interested and Web-savvy DQ visitors to note how to later access DQO for further SLG play at home with family or friends.

Interoperability linking between DQ and DQO—DQ and DQO were co-designed from the beginning to work together. How they were intended to work together did change throughout the development process. This may not be surprising given that we were looking at alternative ways to link both the games and the game play experiences together. The engagement of Creative Kingdoms Inc. help to provide a strategy and technical infrastructure for how to connect the physical game of DQ with the online game play of DQO. As both DQ and DQO would be based on databases, it was then possible to develop a system (server) architecture and data sharing regime whereby user registrations and scores/points could be exchanged between the DQ and DQO databases, as well as providing a security mechanisms (firewall and data replication server) so that either game system could operate, if needed, without the other, but otherwise share common data in a controlled and tractable manner. Finally, it should be noted that another reason for linking DQ and DQO together was to also create in-game incentives for players (e.g., families) who access DQO over the Web to come to the DSC to complete their game play—or more simply, to help increase visitor attendance and membership at DSC. As a result (or maybe as just a coincidence) attendance at DSC has grown significantly since DQ and DQO became operational. This means more families and school groups are coming to DSC, and they are repeatedly visiting DSC at a much higher rate than before DQ and DQO were available.

Discussion and planned enhancements to DQ and DQO

The first topic of discussion is whether DQ and DQO are fun to play. Since fun may be in the mind of the player, we can report results from sustained informal observations (along with a multitude of DSC administered surveys and interviews) that the quick and simple answer is: yes, they are fun to play, but each is fun in a different way. Second, do the players learn anything useful about scientific inquiry or life science? Again based on the same instrumentality for observation, the quick and simple answer is yes, though what is learned in DQ and DQO are different. In both situations, DQ is a game played in a physical venue with other people and involves personal activity and mobility. DQO is a game played in an online, Web-based venue where other people may be involved in play, but their involvement may be centered around interaction at the human-computer interface or through in-game dinosaur characters foraging in a simulated ecological niche. So we expect that what's fun and what's learned will be different, but we continue to seek to understand how and why they are different.

The physical DQ venue and game play experience is difficult to upgrade, while the DQO venue (at least the game engine software, game content, and back-end database services) can be more readily upgraded, as resources become available. Accordingly, we are investigating a number of planned upgrades to DQO that will further expand the collaboration affordances and game play experiences that can be realized more readily in an online, Web-based game platform.

First, reflecting the diversity of people (students, parents, and others) who visit DSC, we seek to provide multi-lingual game play user interfaces for DQO in languages such as Spanish and Korean. Actually, we developed DQO with internationalization and localization as part of its design and implementation scheme, so provision of multi-lingual support is primarily one of adding/replacing corresponding in-game textual content across languages.

Second, following from this, we seek to provide multi-national deployments for DQO to non-English speaking venues. Dinosaur themed interactive science exhibits are being developed in areas like Mexico, Latin America, and South Korea. Though DQO was designed with California Science Education Standards in mind, our correspondence with colleagues in those areas indicates such standards are acceptable as a starting point.

Third, we seek to expand the multi-player capabilities of the DQO DinoSphere to accommodate more MMOG services and game play modes. Our intent is to add such capabilities to the highest levels of DQO DinoSphere game play, so that existing game content and play experience will be minimally affected. The wisdom of this choice remains to be seen and realized.

Last, our original goals included making DQO a networked SLG environment that could be built from open source software, support open content artwork, and accommodate a controlled interface to the Web of open and current scientific research related to paleontology and paleobiology. None of this has been realized, and the technical choices that we have invested will limit what we can do to realize these goals. Nonetheless, they remain part of our long-term goal, and we look forward to opportunities that will enable to develop more collaborative SLGs in other scientific domains and for other public audiences who want informal science education experiences and resources.

Conclusions

In this paper, we described some of the concepts and results that arose from the development and deployment of a large-scale collaborative game environment called DinoQuest and DinoQuest Online. As an interoperable game environment, DQ and DQO provide a unique experience and approach to informal science education, as well as matching California/National Science Education Standards. As noted, games for helping school-age children to learn about science (or more specifically, life science) is not a focus of the global computer game industry, yet we believe it represents an important and under-served community of potential game players and others (parents, teachers) who want to informally collaborate in and around such a science learning game environment. Subsequently, we identified and examined different collaborative group forms that emerged to play DQ and DQO, as well as the affordances that help facilitate collaborative game play. Along the way we provided examples of the collaborative groups, affordances, and game play from DQ and DQO.

In closing, we welcome participants of the International Workshop on Collaborative Games to either make their way from Irvine to Santa Ana to visit the Discovery Science Center (about a 15-30 minute drive from the CTS 2008 Conference Hotel, depending on traffic) to see and experience DinoQuest, or to register and play one or more of the DinoQuest Online game modules found starting at http://www.DQOnline.org. Both DQ and DQO will provide different but comparable experiences for collaborative games and game play, each of which can last for 3-5 hours.

Acknowledgements: Development of DinoQuest and DinoQuest Online was supported by the Discovery Science Center, its members, and many governmental and corporate sponsors. Recent R&D projects at the UCGameLab are supported with contracts and grants from the Digital Industry Promotion Agency (DIP) in Daegu, South Korea, Intel Research, California Institute for Telecommunications and Information Technology (Calit2), and others. No endorsement implied. Some of the other participants involved in the development of DinoQuest and DinoQuest Online include Creative Kingdoms, Inc., Alex Szeto (DQO game art and programming), Calvin Lee (DQO database programming), Janet Yamaguchi (California Science Education Standards, and Education Programs at DSC), and Celia Pearce (design contributions).

References

Allison-Bunnell, S. and Schaller, D.T. (2005). From the Physical to the Virtual: Bringing Free-Choice Science Education Online, in L. Tan and R. Subramanian (eds.), *E-Learning and Virtual Science Centers*, 163-189, Hershey, PA, IGI Global Publishing, 2005.

Bainbridge, W.S. (2007). The Scientific Research Potential of Virtual Worlds, *Science*, 317, 472-476.

Bogost, I. (2007). *Persuasive Games: The Expressive Power of Videogames*, MIT Press, Cambridge, MA.

Collabs, (2008). The Science of Collaboratories, http://www.scienceofcollaboratories.org/

DSC, (2008). Discovery Science Center, Santa Ana, CA. http://www/discoverycube.org

Gee, J.P. (2003). *What Computer Games Have To Teach Us About Learning and Literacy,* Palgrave Macmillan, New York.

Koster, R. (2004). *A Theory of Fun for Game Design*, Paraglyph Press.

LaFarge, A. and Nideffer, R. (2002). Shift-Ctrl: Computer, Games, and Art, *Leonardo: A Journal of Art, Science, and Technology.* MIT Press. 35(1), 5-13, Winter.

Malone, T. and Lepper, M.R. (1987). Making Learning Fun: A taxonomy of intrinsic motivations for learning, in R.E. Snow and M.J. Farr, *Aptitude, Learning, and Instruction III: Conative and Affective Process Analyses*, 223-253, Erlbaum, Hillsdale, NJ.

Nardi, B. and Harris, J. (2006). Strangers and Friends: Collaborative Play in World of Warcraft, *Proc. 2006 Conf. Computer Support Cooperative Work (CSCW'06)*, 149-158, Banff, Canada, ACM Press.

Nideffer, R. (2002). *PROXY*, The Whitney Museum Portal to Net Art, http://artport.whitney.org/exhibitions/biennial2002/nideffer.shtml.

Nideffer, R. (2006). *unexceptional.net*, http://transliteracies.english.ucsb.edu/post/research-project/project-members/robert-nideffer

Nideffer, R. (2007). Game Engines as Embedded Systems, in V. Vesna (ed.), *Database Aesthetics: Art in the Age of Information Overflow.* University of Minnesota Press. Minneapolis, Minnesota.

Salen, K. and Zimmerman, E. (2003). *Rules of Play: Game Design Fundamentals*, MIT Press.

Scacchi, W. (2004). Free/Open Source Software Development Practices in the Computer Game Community, *IEEE Software*, 21(1), 59-67, January/February 2004.

Scacchi, W. and Adams, J. (2007). Recent Developments in Science Learning Games for Informal Science Education, presentation at *Games, Learning, and Society: 3.0*, Madison, WI. July.

Shaffer, D. W. (2006). *How Computer Games Help Children Learn*, Palgrave Macmillan, New York.

Teasley, S., & Wolensky, S. (2001) Scientific collaborations at a distance. *Science*, 292, 2254-2255.

UCGameLab, (2008). Game Culture and Technology Laboratory, University of California, Irvine, Irvine, CA. http://www.UCGameLab.net

WDIL (2008). *Web Designs for Interactive Learning*, http://www.wdil.org/

Collaboration tools and techniques for dispersed teams of players/developers

Virtual Collaboration Portal (VCP)

As part of our efforts on the DVAS project, we have been developing a prototype for a second generation Virtual Collaboration Portal (VCP). The current status of our efforts can be viewed at http://vcp.ics.uci.edu.

An earlier implementation of the VCP can be found at: http://proxy.arts.uci.edu/package/content.php?ctID=1 (see Figure 1). The portal was implemented as a custom built configuration management system (CMS) on top of PHP and MySQL. It supported many basic features such as: user login/profile, members list, user to user messages, groups, tags, user rank based on usage patterns, and layout templates. In addition, several third party tools were integrated in the VCP, including: wiki (Mediawiki), blogs (B2Evolution), a news feeder (Gregarious), version control (DVCS), teleconferencing (Flash/PHP Chat), a search engine/webcrawler (Lucene/Nutch).

This implementation proved to be brittle due to API and structural evolution of a myriad of integrated external tools. Maintenance became prohibitively expensive and several features have been lost over time.

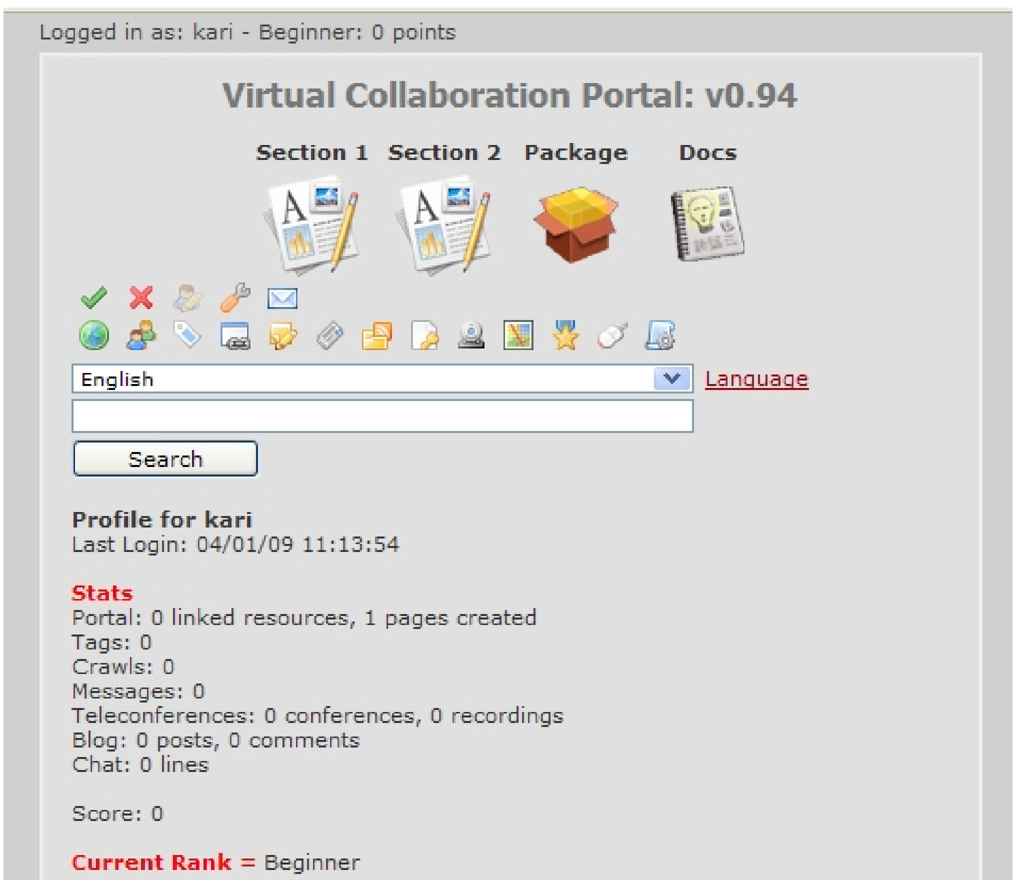


Figure 1: Previous implementation of VCP

Next Generation VCP

For the next generation VCP we decided to leverage off an Open Source CMS with many of the required features built in. We have chosen Drupal, a content management framework geared towards configurability and customization. Some of the advantages to using Drupal as a development platform include a very large community (> 450K user accounts, > 2K developers), thousands of user contributed modules (> 3700), a leader in market share (top three, along with Wordpress and Joomla). Also, a somewhat unique feature of Drupal is the ability to create any number of user roles and to grant fine grain permissions to users based on those roles.

An initial version of a Drupal-based prototype of the VCP is pictured in Figure 2. In the remainder of this section we describe the current features of our prototype.

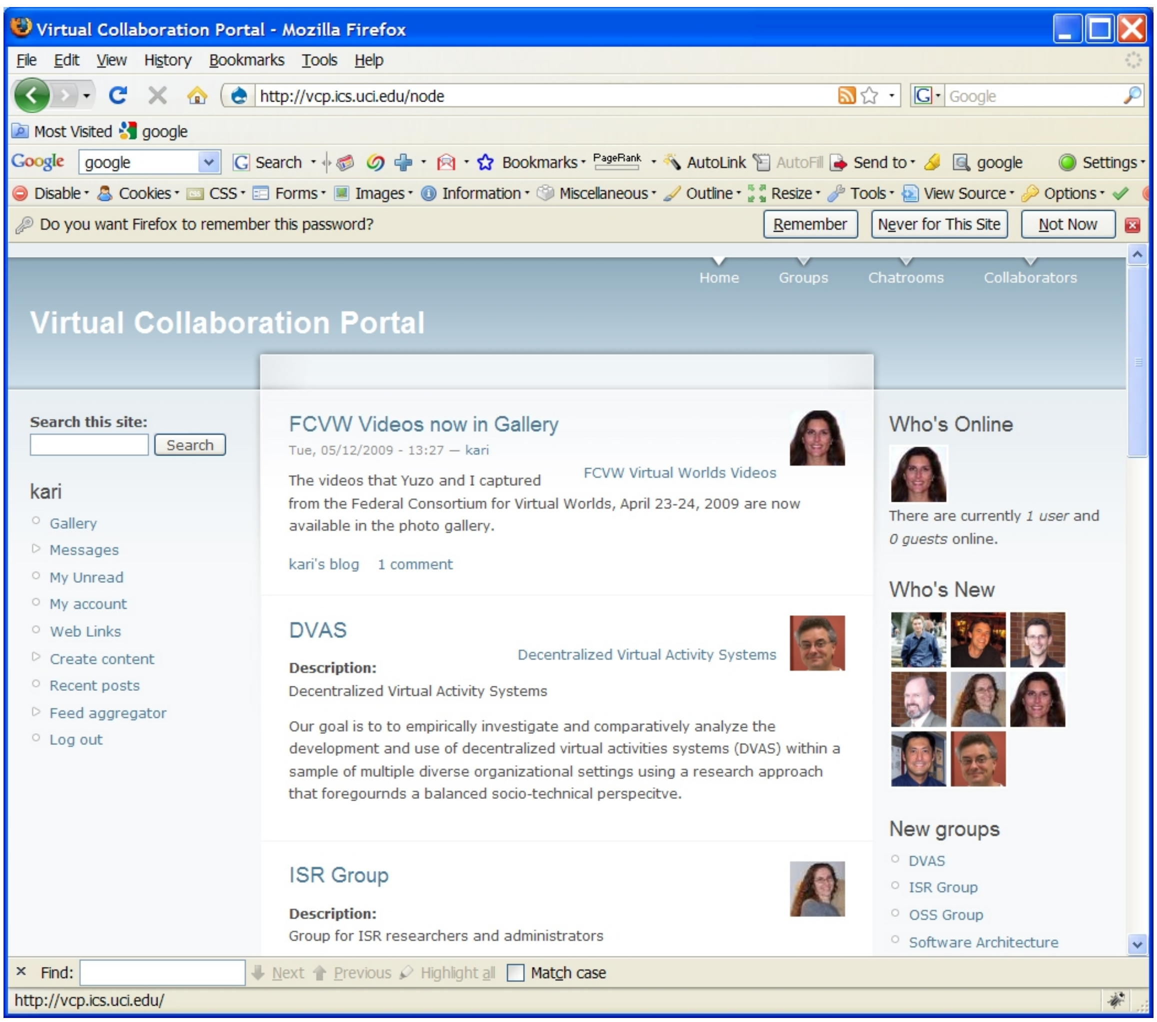


Figure 2: Next generation, Drupal-based VCP

User Login/Accounts

The users in the VCP are called *collaborators*. Figure 2 shows the home page of the VCP for a logged in collaborator. The home page includes a listing of the latest VCP posts visible to the given collaborator. Each entry shows the image of its author and that image links to the profile page for that collaborator. Primary links are displayed in the upper right corner of the page, and additional navigation links are provided in the left column. Collaborators can see images of other collaborators currently logged in displayed in the upper right column. Below that is a block of images showing the most recent collaborators to join the portal.

Figure 3 shows the VCP login page. Notice that the navigation links are limited and that the user can only see public posts. Anonymous users cannot access profile pages. From here the user can either login or create a new account on the portal.

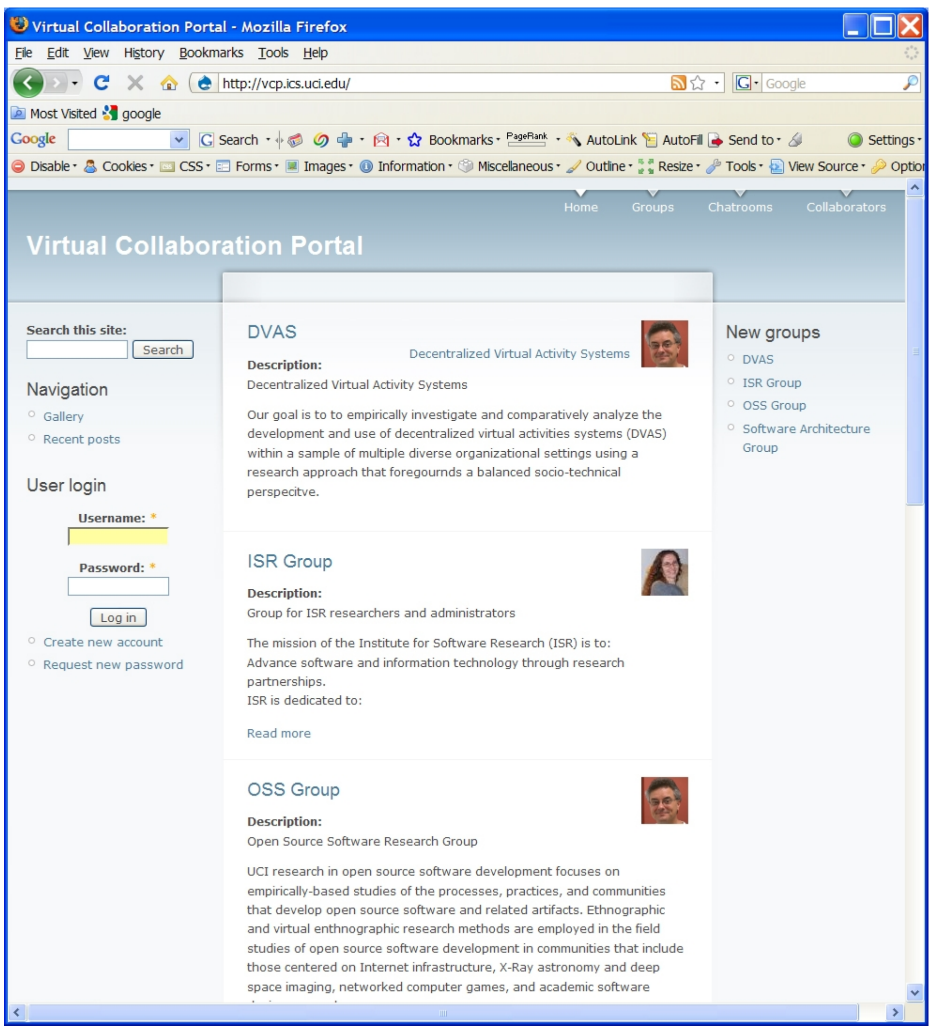


Figure 3: VCP Login Page

Figure 4 shows the creation of a new user account. The fields marked in yellow (username, email, and full name) are mandatory. Users are invited to join any "open" groups (see Content) at this time. The portal prevents spam by using a module that integrates Drupal with the reCAPTCHA anti-bot service to ensure that a human is interacting with the portal.

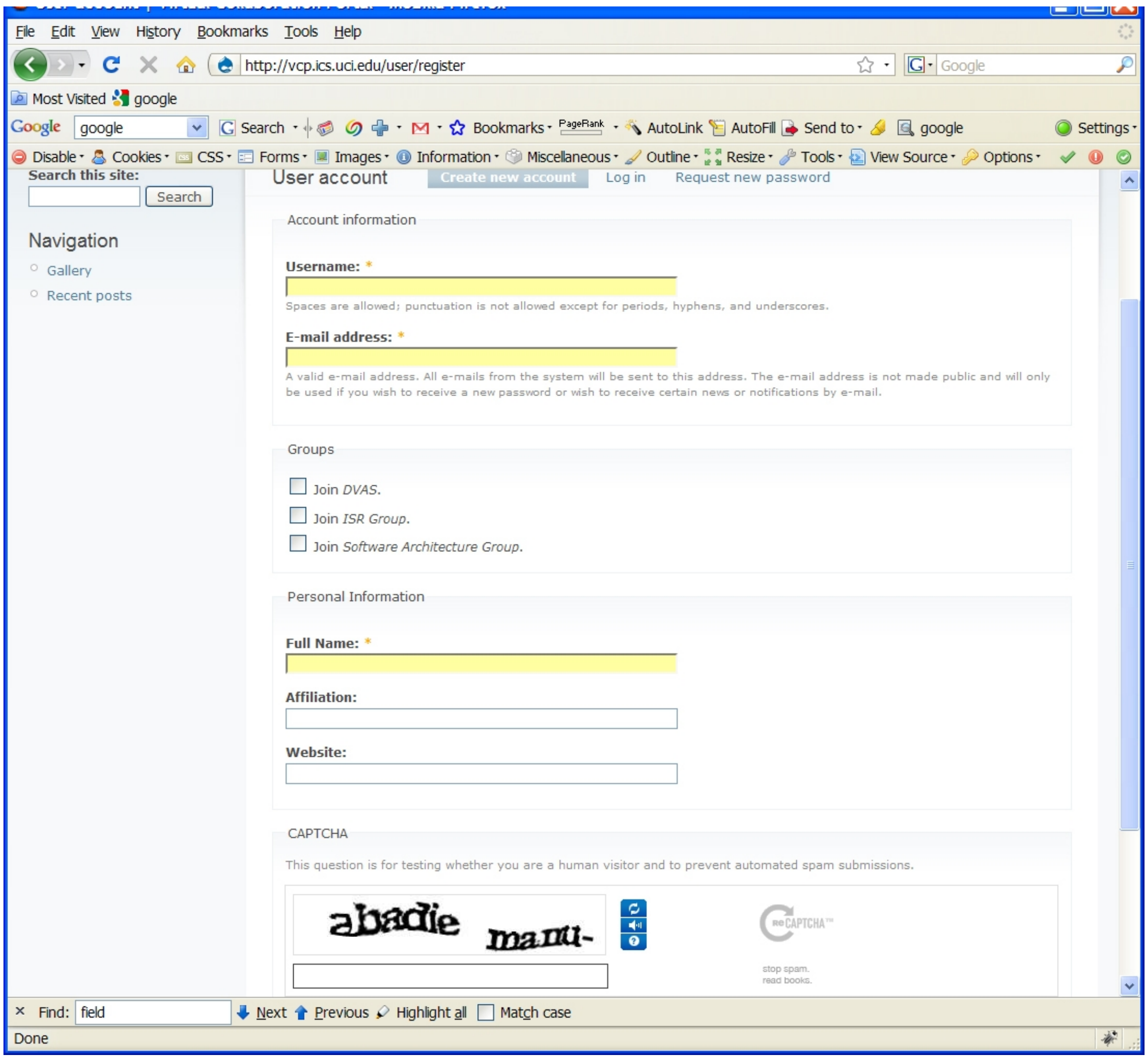


Figure 4: Creating a new user account.

Collaborators and Profiles

One of the primary links, *Collaborators*, brings up a searchable page that lists all collaborators in the system, including their names, images, and affiliations. This page is shown in Figure 5. Each name and image links to a profile page for that collaborator.

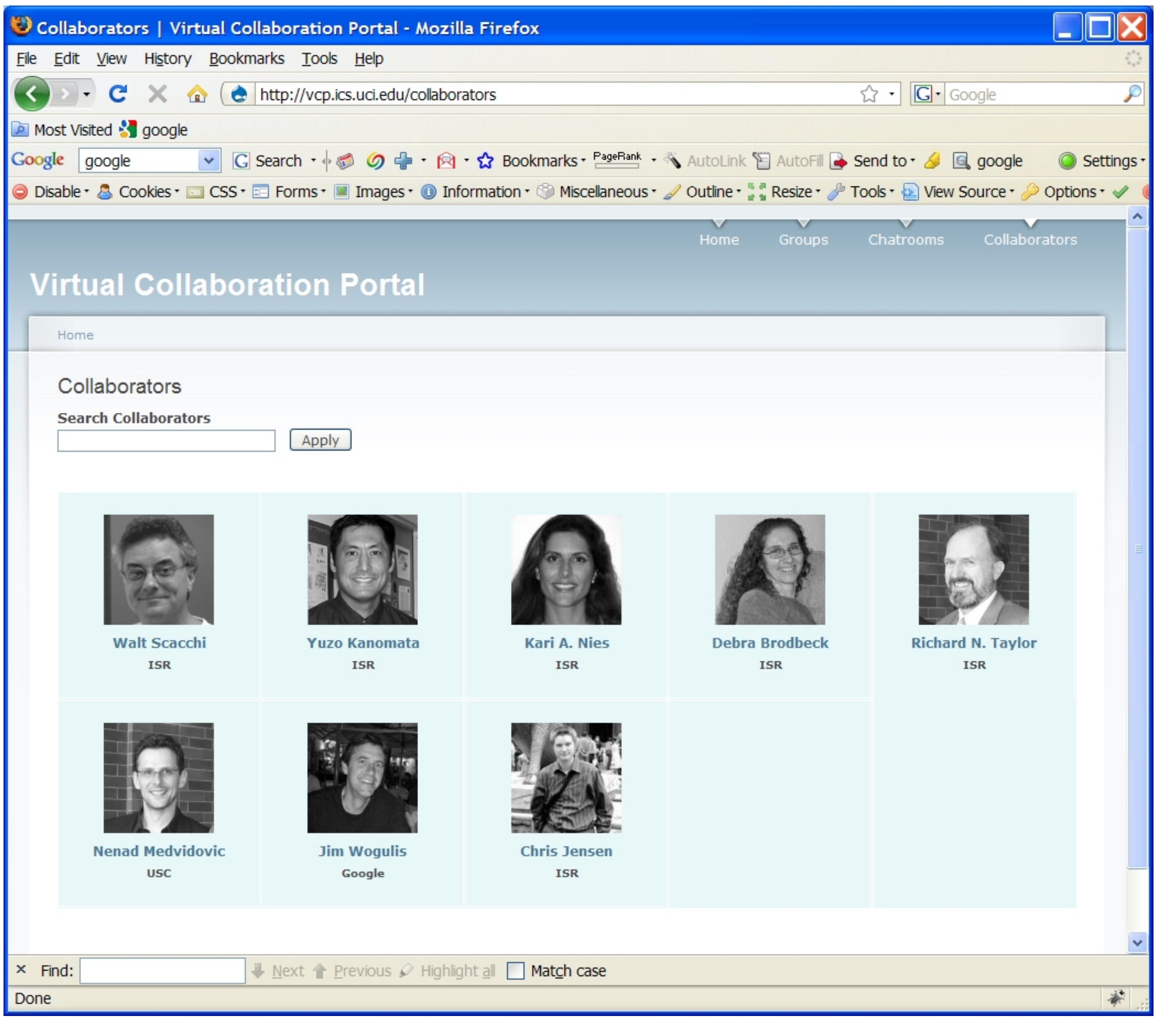


Figure 5: VCP Collaborators

A collaborator's profile is shown in Figure 6.. In this example, Kari is viewing Walt's profile page. The profile page shows the collaborator's full name, affiliation, and image.

Collaborators can also include a link to their professional or personal website. In addition, all of the groups to which the collaborator is affiliated are listed (see Content). User Kari has already subscribed to receive notifications of any posts by Walt (see below). She can cancel that subscription here. There is also a link to all of Walt's recent blog postings.

From here a collaborator can also "Track" all content (see Content) belonging to Walt and view any "Web Links" (see) that he has created.

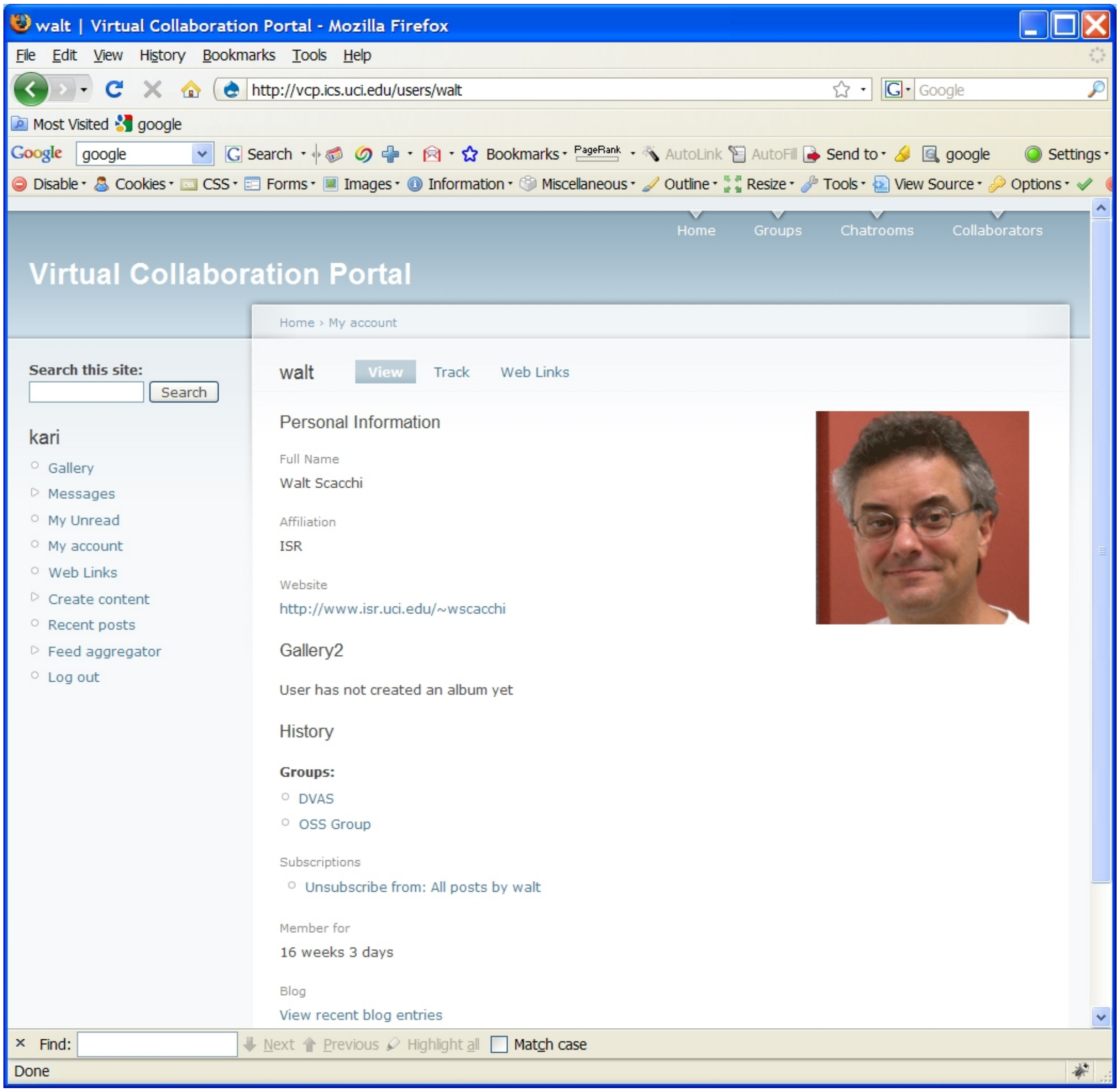


Figure 6: A Collaborator's Profile

Once a collaborator has logged into portal he or she can access and edit their account by following the “My Account” navigation link in the left column. A collaborator’s view of their own profile is shown in Figure 7. Here the user can change their username, email address, password, or upload a new image. They can also follow the “Personal Information” tab to edit the information visible in their profile. From here they can manage their subscriptions (see below), view the notifications received from those subscriptions, and track their own content.

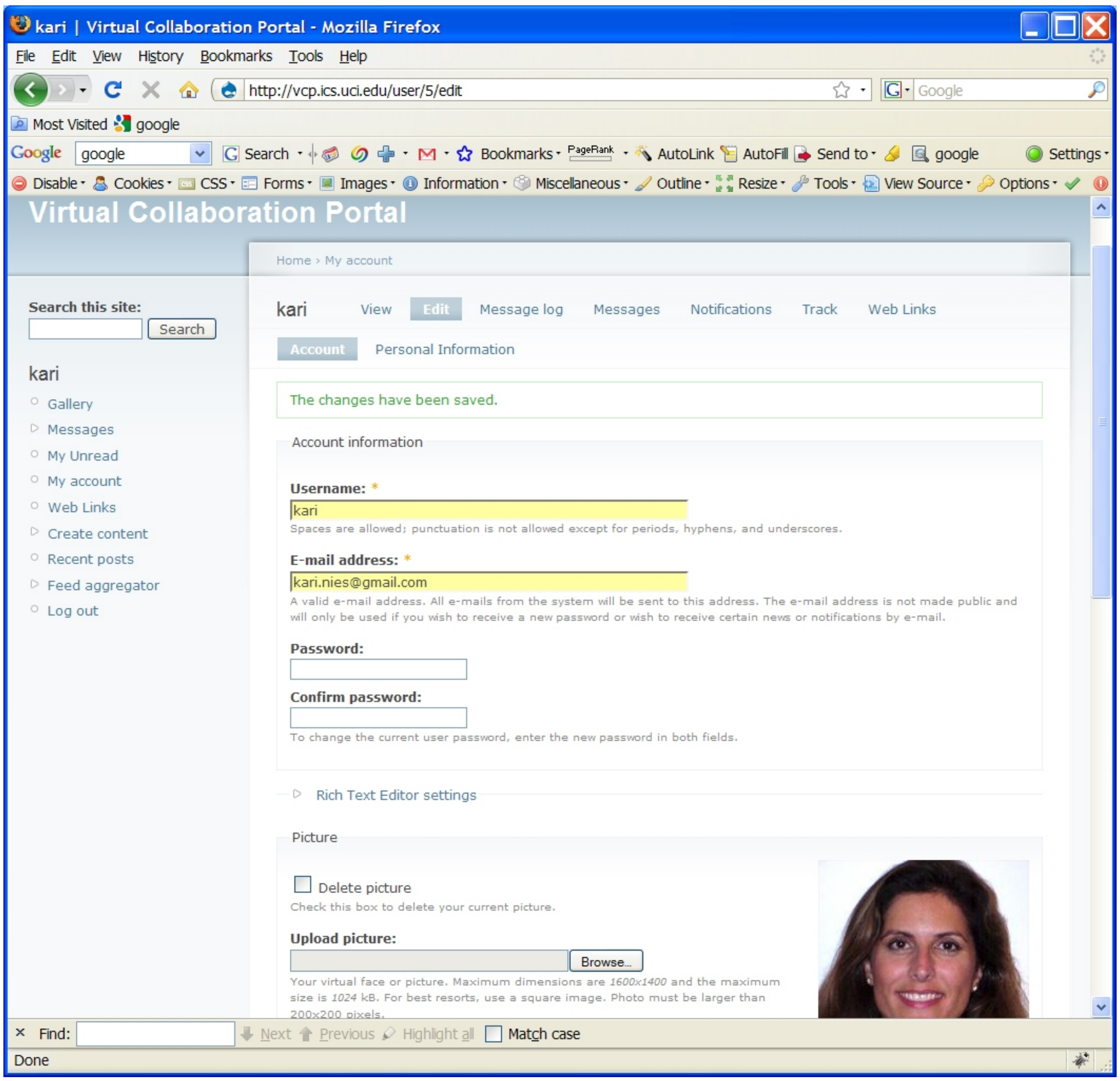


Figure 7: Collaborator's view of their own account.

Content

Collaborators can publish a variety of content types using the "Create content" navigation menu in the left column. Figure 8 shows the available content types. Blog entry, page and story are part of the Drupal core. Chat rooms and Web Links are discussed below in and respectively. A custom content type was implemented to support video upload with embedded display. Video content is described in more detail below.

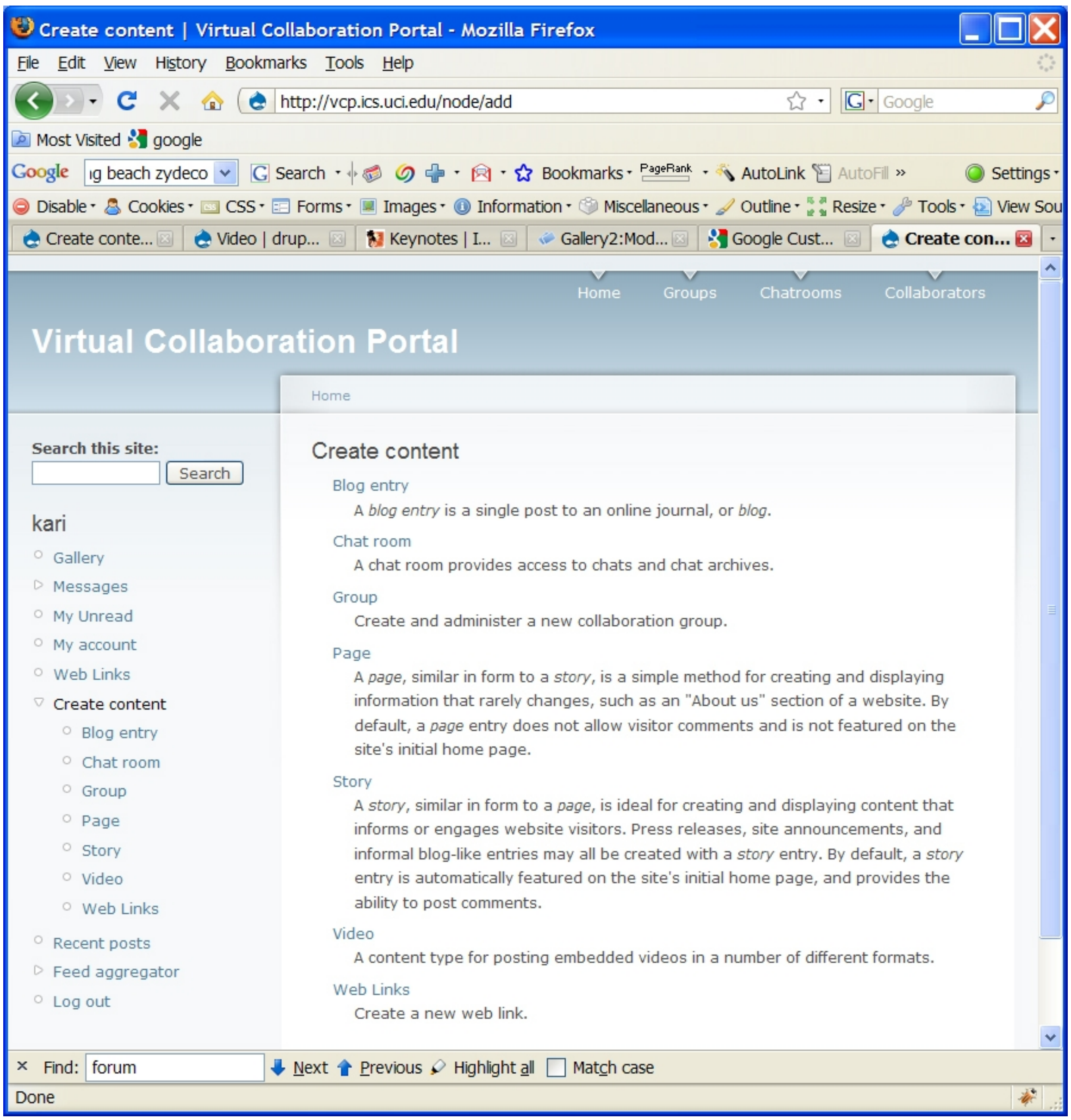


Figure 8: Various content types available to collaborators.

Figure 9 gives an example of how a collaborator can create content in the VCP. In this example the user is creating video content which allows the uploading and display of embedded video in various formats. Only the most pertinent fields are described in this example. The user specifies a title, tags and body for the video content. Most Drupal content contains both a title and a body. Here the body is simply a description of the video content. In addition, we have added the concept of Tags to the VCP. Tags are keywords added by the content creator that can be used for searching and receiving notifications about content.

This content type has been group-enabled (see Groups), therefore the creator can choose to share this content with any groups to which they belong. By selecting the Public checkbox the content becomes visible to all Collaborators. Here the creator chooses to allow fellow collaborators to both read and write comments about the content. And finally, the creator can browse his local file system to specify a path to the video file which is to be uploaded to the VCP.

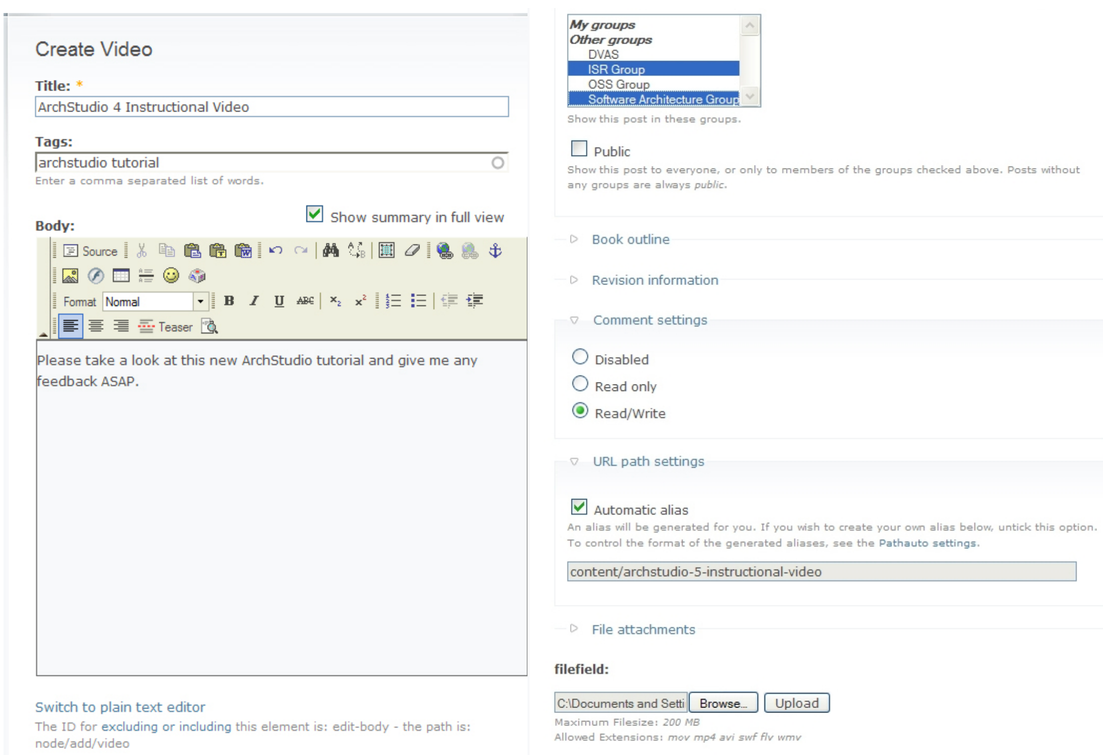


Figure 9: Creating video content

Figure 10 shows the created video content within the VCP. Note that collaborators have created comments and that these comments can be threaded as shown in this example.

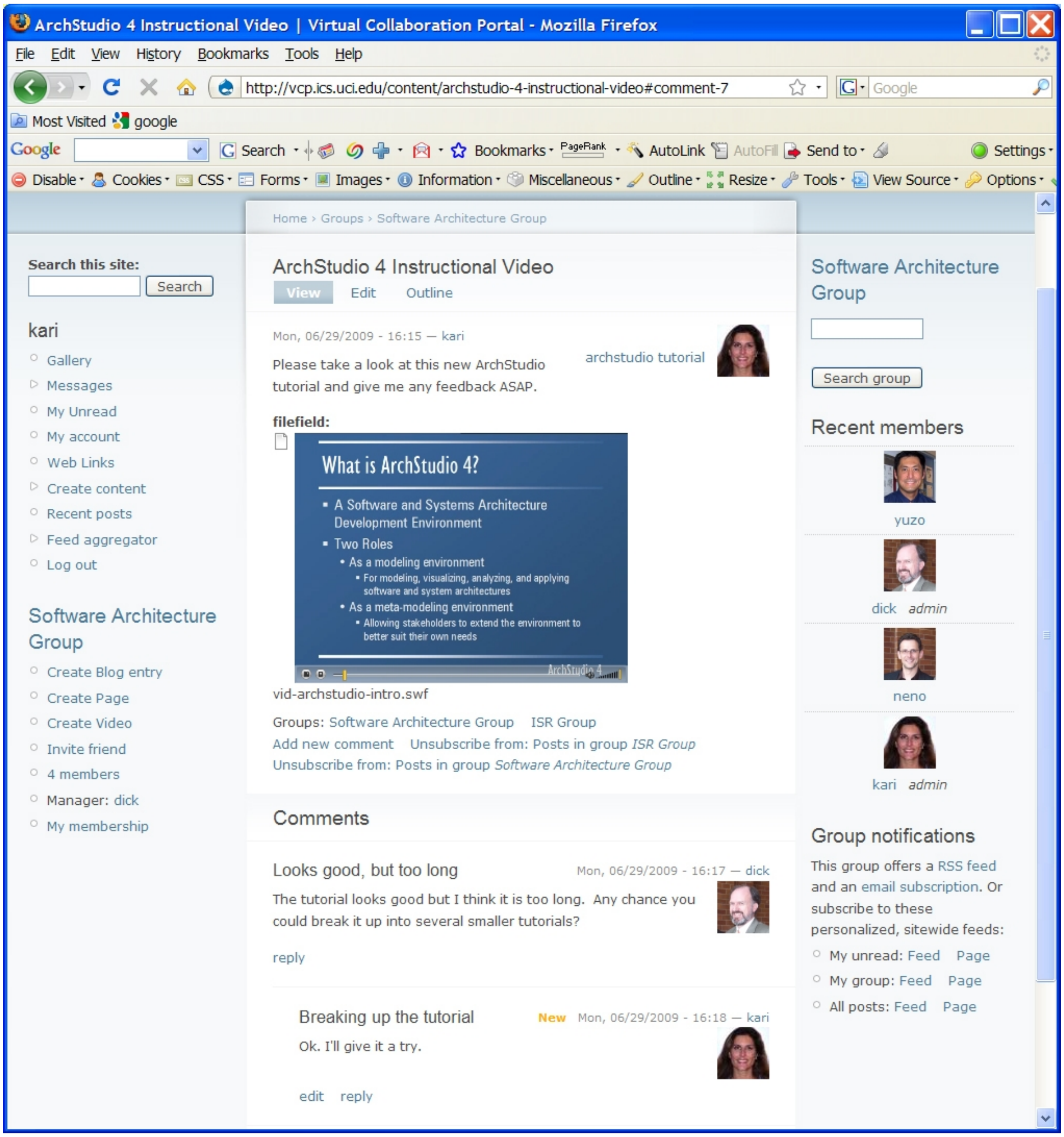


Figure 10: Posted video content with comments

Groups

The VCP allows collaborators to create and manage their own ‘groups’ to which other collaborators can subscribe. Each group maintains its own homepage where collaborators can communicate amongst themselves. Figure 11 shows the homepage for the DVAS group. Subscribers can create new content, invite collaborators to join the group, view a full list of members, find the manager, or unsubscribe to the group using the navigation menu in the lower left column. The center screen shows a list of recent postings to this group. In the right column a subscriber can search the content from this group. Some of the most recently added members are displayed below that. RSS feeds and email subscriptions are also available for groups. For non-subscribers, a group homepage has a restricted view as shown in Figure 12.

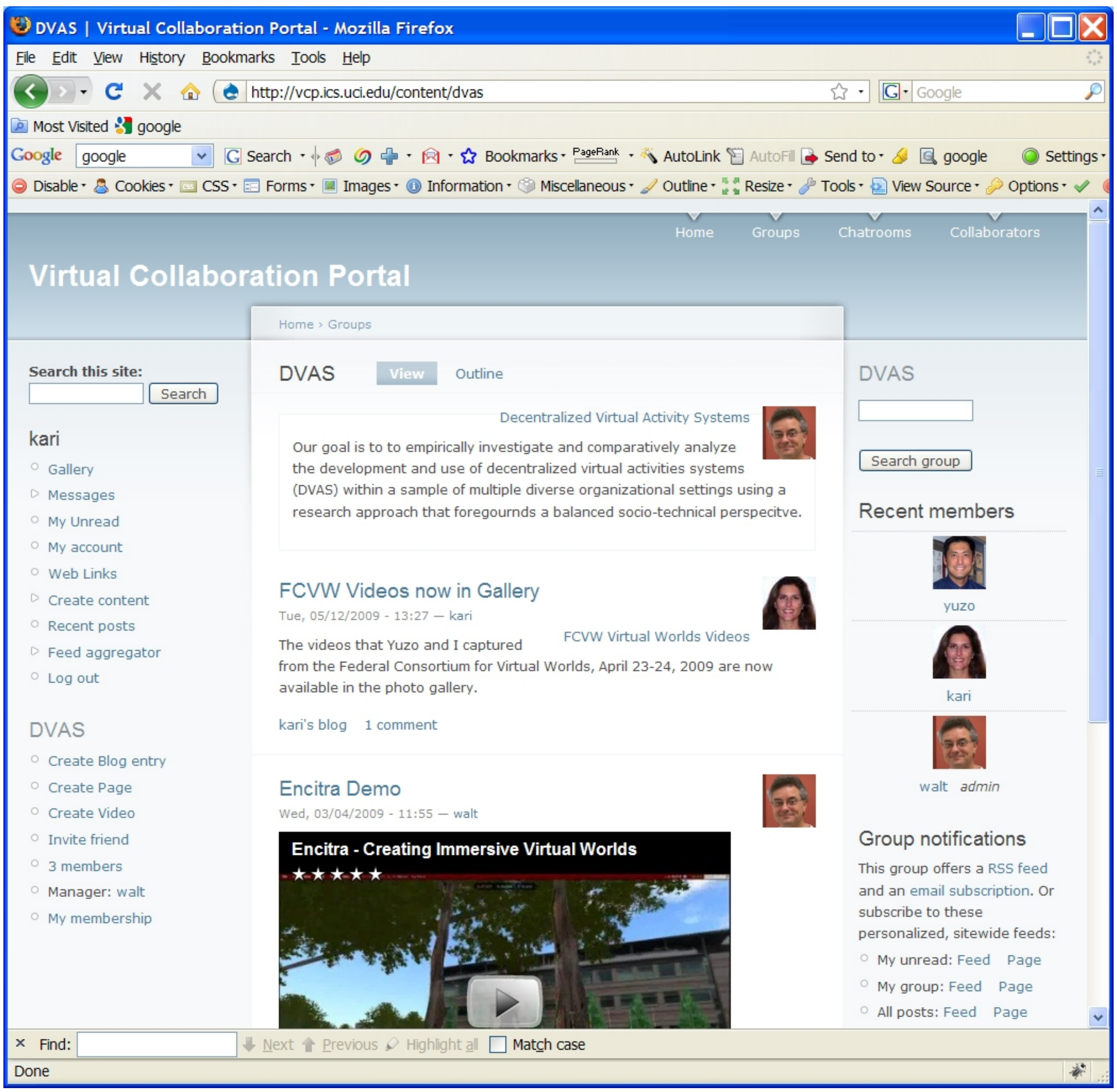


Figure 11: DVAS group page, subscriber

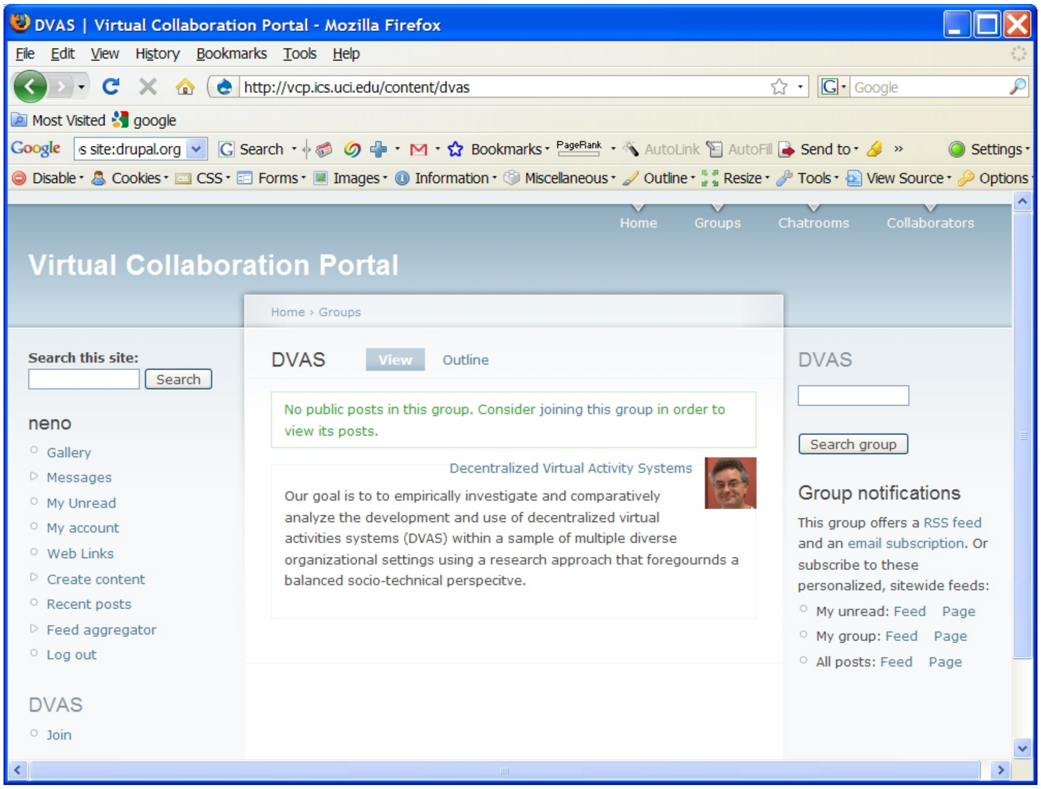


Figure 12: DVAS group page, non-subscriber

Collaborators can view get a summary of all content that they have not yet viewed by selecting "My Unread" in the navigation menu, as show in Figure 13.

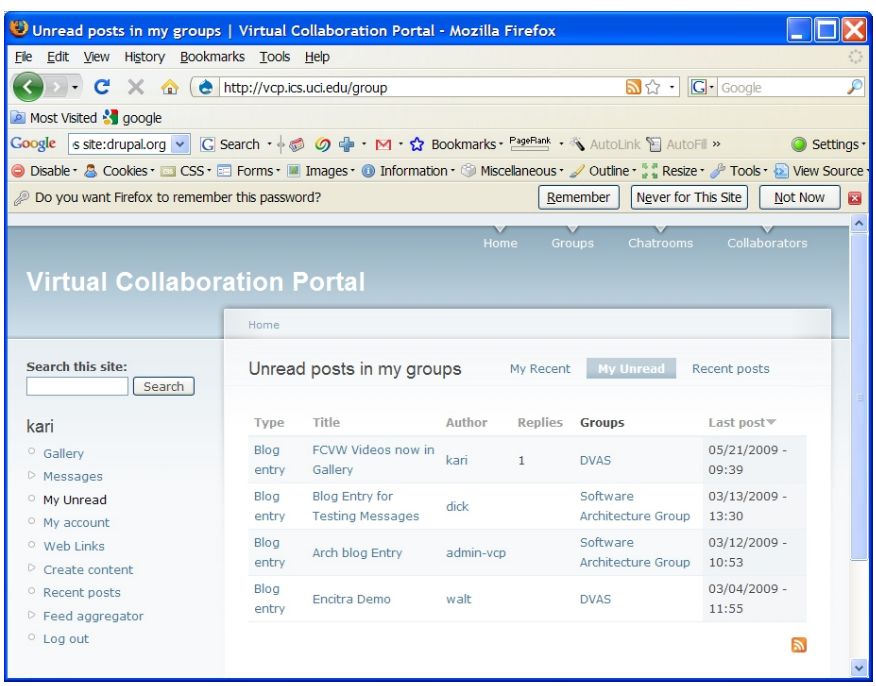
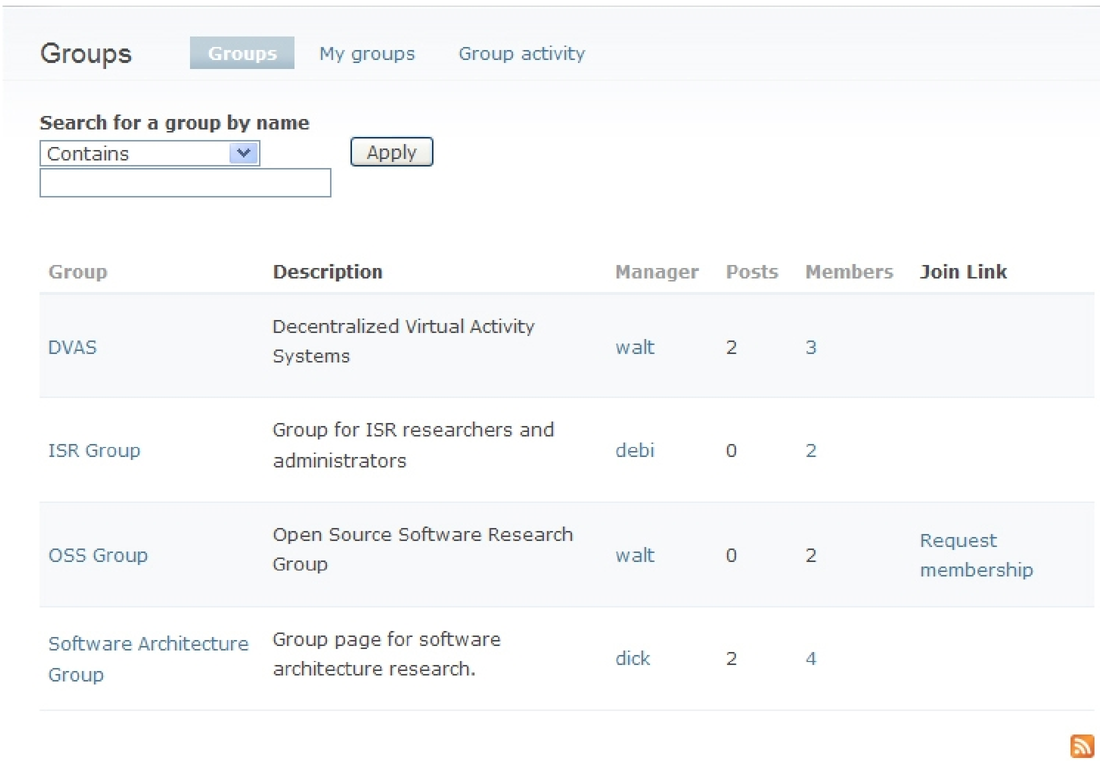


Figure 13: View of unread posts

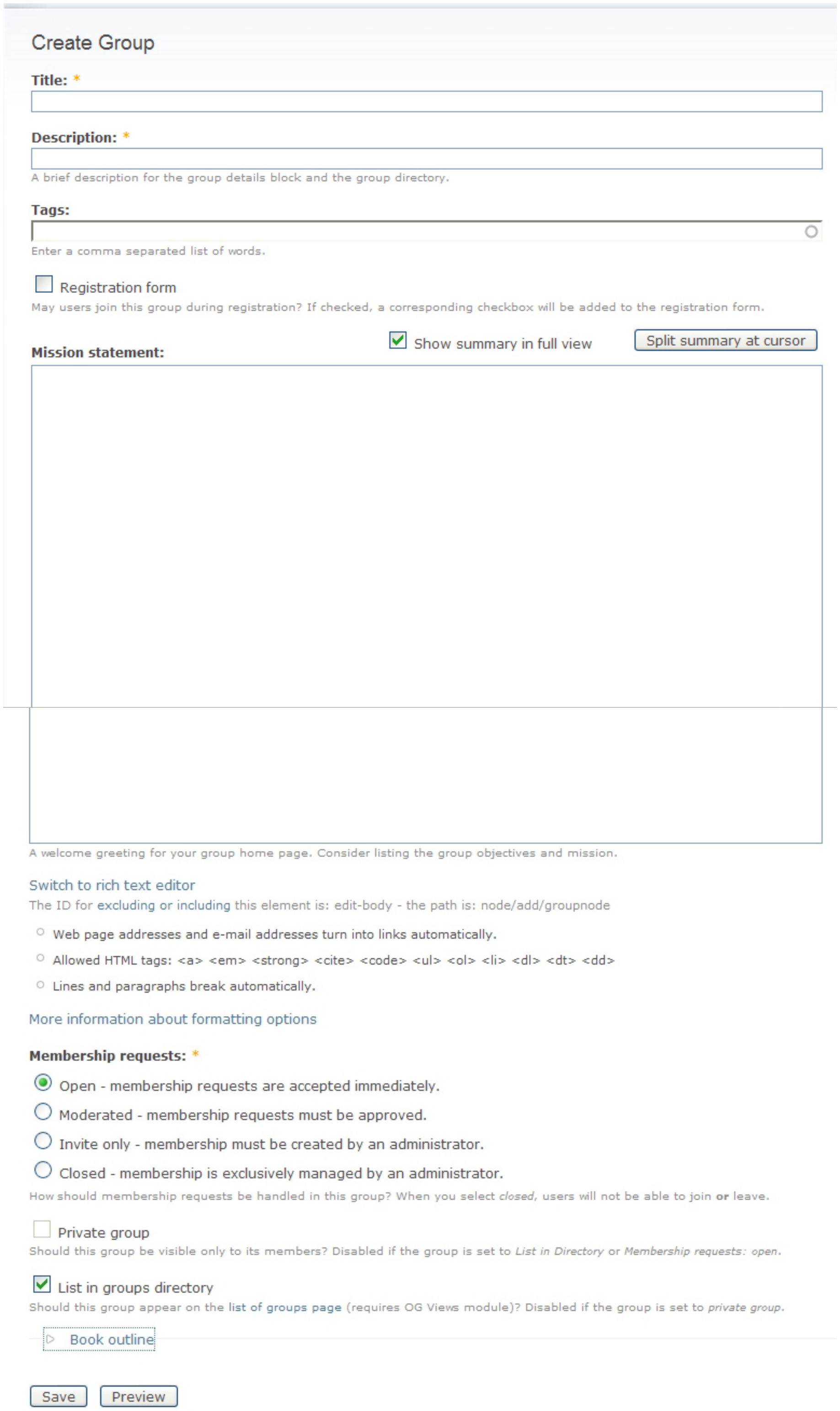
A group directory listing can be accessed by following the "Groups" link in the primary menu found in the upper right corner of the all pages. A sample listing is given in Figure 14. An RSS feed is available for this listing so that Collaborators can monitor new groups.



Group	Description	Manager	Posts	Members	Join Link
DVAS	Decentralized Virtual Activity Systems	walt	2	3	
ISR Group	Group for ISR researchers and administrators	debi	0	2	
OSS Group	Open Source Software Research Group	walt	0	2	Request membership
Software Architecture Group	Group page for software architecture research.	dick	2	4	

Figure 14: Groups directory

Figure 15 shows the creation of a new group. When collaborators create new groups they must provide a group name (Title), a brief description for the group listing, keywords (Tags) that describe the group, and a mission statement. The creator can indicate whether on or not users should have the opportunity to join the group at the time they create an account. There are several privacy options to choose from. The creator must indicate if this group is Open - any registered collaborator can join at any time, Moderated - all membership request must be approved by an administrator, Invite-only - members must accept an invitation initiated by an administrator, or Closed – only administer can create a membership. The creator can decide if the group should be listed in the Group Directory and the group can be set to private (provided it is not Open and unlisted) which makes visible to only its members.



Create Group

Title: *

Description: *

A brief description for the group details block and the group directory.

Tags:

Enter a comma separated list of words.

Registration form

May users join this group during registration? If checked, a corresponding checkbox will be added to the registration form.

Mission statement:

Show summary in full view

Split summary at cursor

A welcome greeting for your group home page. Consider listing the group objectives and mission.

Switch to rich text editor

The ID for excluding or including this element is: edit-body - the path is: node/add/groupnode

- Web page addresses and e-mail addresses turn into links automatically.
- Allowed HTML tags: <a> <em> <strong> <cite> <code>   <dl> <dt> <dd>
- Lines and paragraphs break automatically.

More information about formatting options

Membership requests: *

- Open - membership requests are accepted immediately.
- Moderated - membership requests must be approved.
- Invite only - membership must be created by an administrator.
- Closed - membership is exclusively managed by an administrator.

How should membership requests be handled in this group? When you select *closed*, users will not be able to join **or** leave.

Private group

Should this group be visible only to its members? Disabled if the group is set to *List in Directory* or *Membership requests: open*.

List in groups directory

Should this group appear on the list of groups page (requires OG Views module)? Disabled if the group is set to *private group*.

Book outline

Save

Preview

Figure 15: Creating a new group

Subscriptions

Collaborators can receive a number of messages from the system based on subscriptions. These messages can be viewed from the collaborators' own account by selecting the "Messages" tab (see an example in Figure 16).

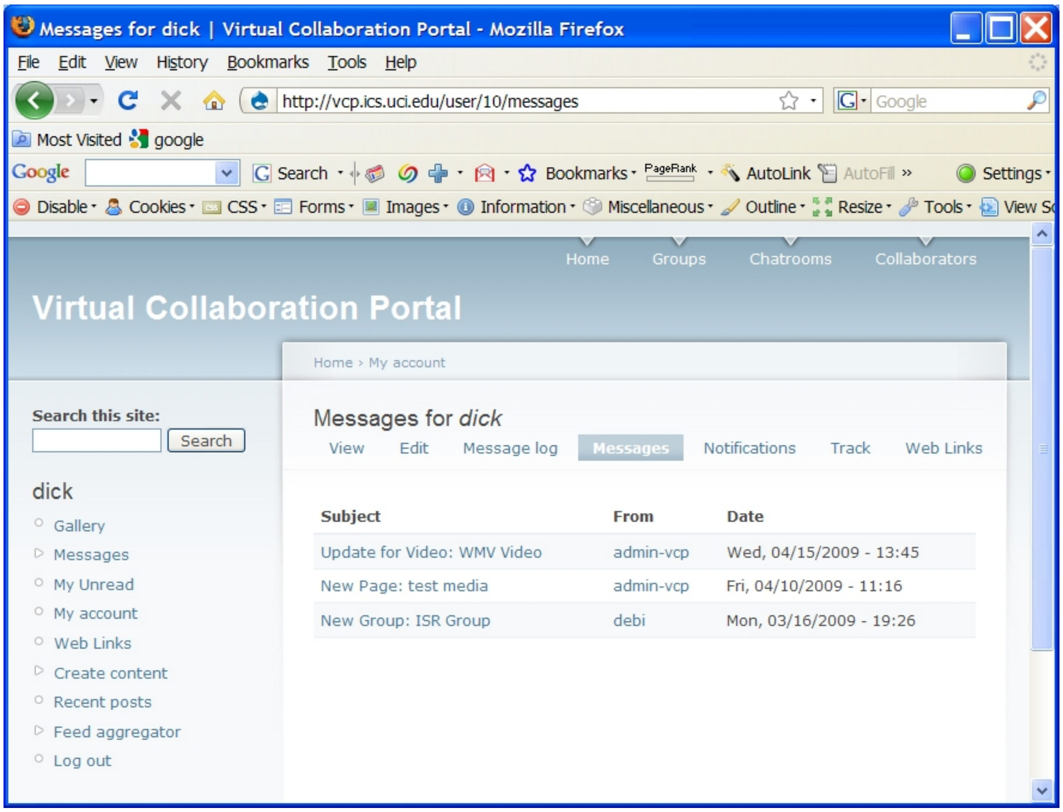


Figure 16: Subscription-based system messages

Collaborators can view their current subscriptions and define new subscriptions by selecting the "Notifications" tab in their account. Notification subscriptions can be based on author, groups, tags, or specific threads. Figure 17 shows the notifications overview. From here you can edit your default subscription settings, temporarily suspend or cancel all subscriptions, and create new subscriptions. An examples of group and author based subscriptions are found in Figure 18. Collaborators can specify the interval at which they wish to receive notifications (Immediately, Every hour, Twice a day, Daily, or Weekly) as well as the delivery mechanism (Web or email).

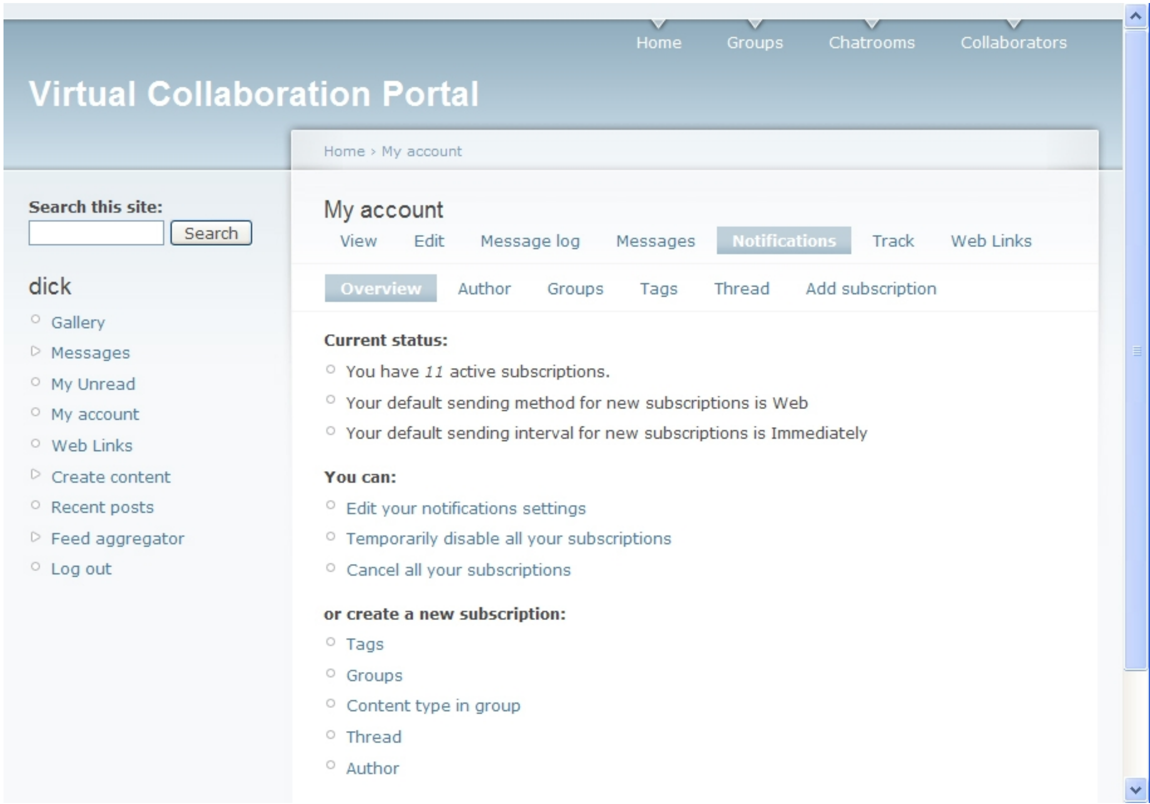


Figure 17: Notifications overview

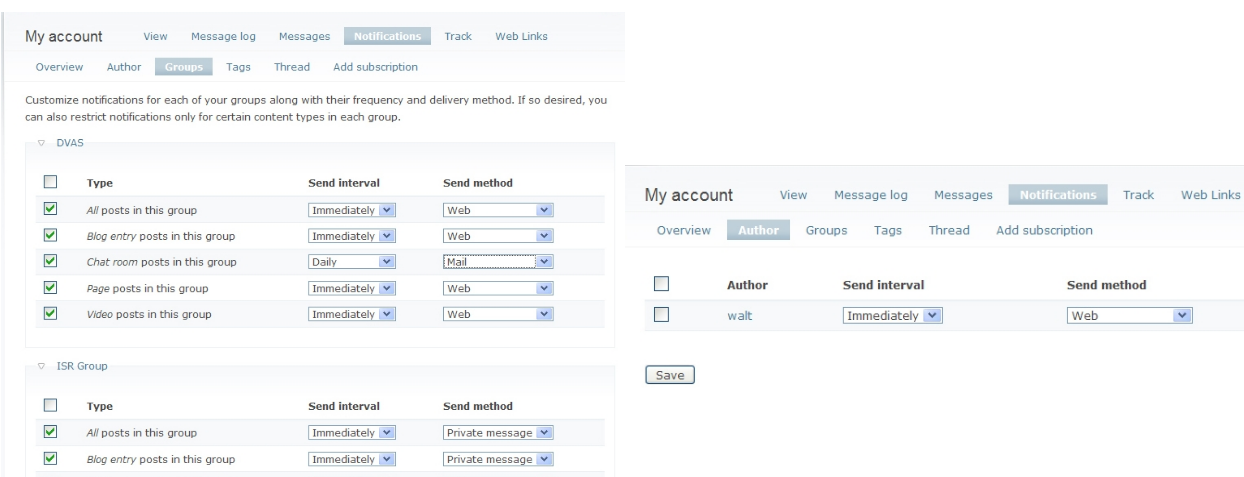


Figure 18: Example subscriptions

User to User Communication

The VCP supports direct communication between collaborators via private messages and a chat system. Direct message between collaborations can be viewed by selecting "Messages" from the navigation menu in the left column. Figure 19 shows a listing of private messages. Unread messages are shown in boldface. An individual message can be seen in Figure 20. From here users can reply to message threads, follow links from the senders name or photo to their profile, delete messages, or block future messages from a give author.

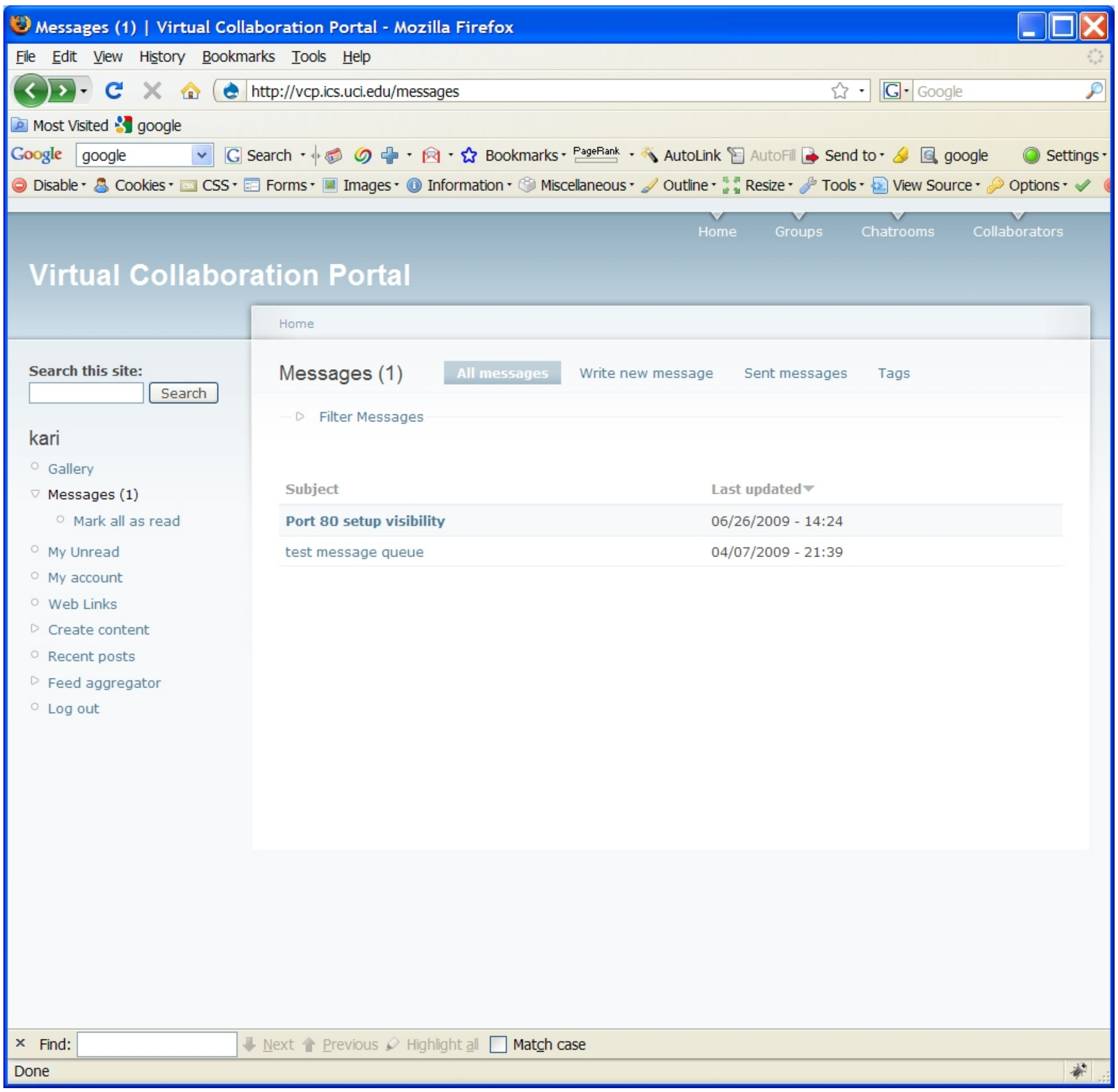


Figure 19: Listing of private messages

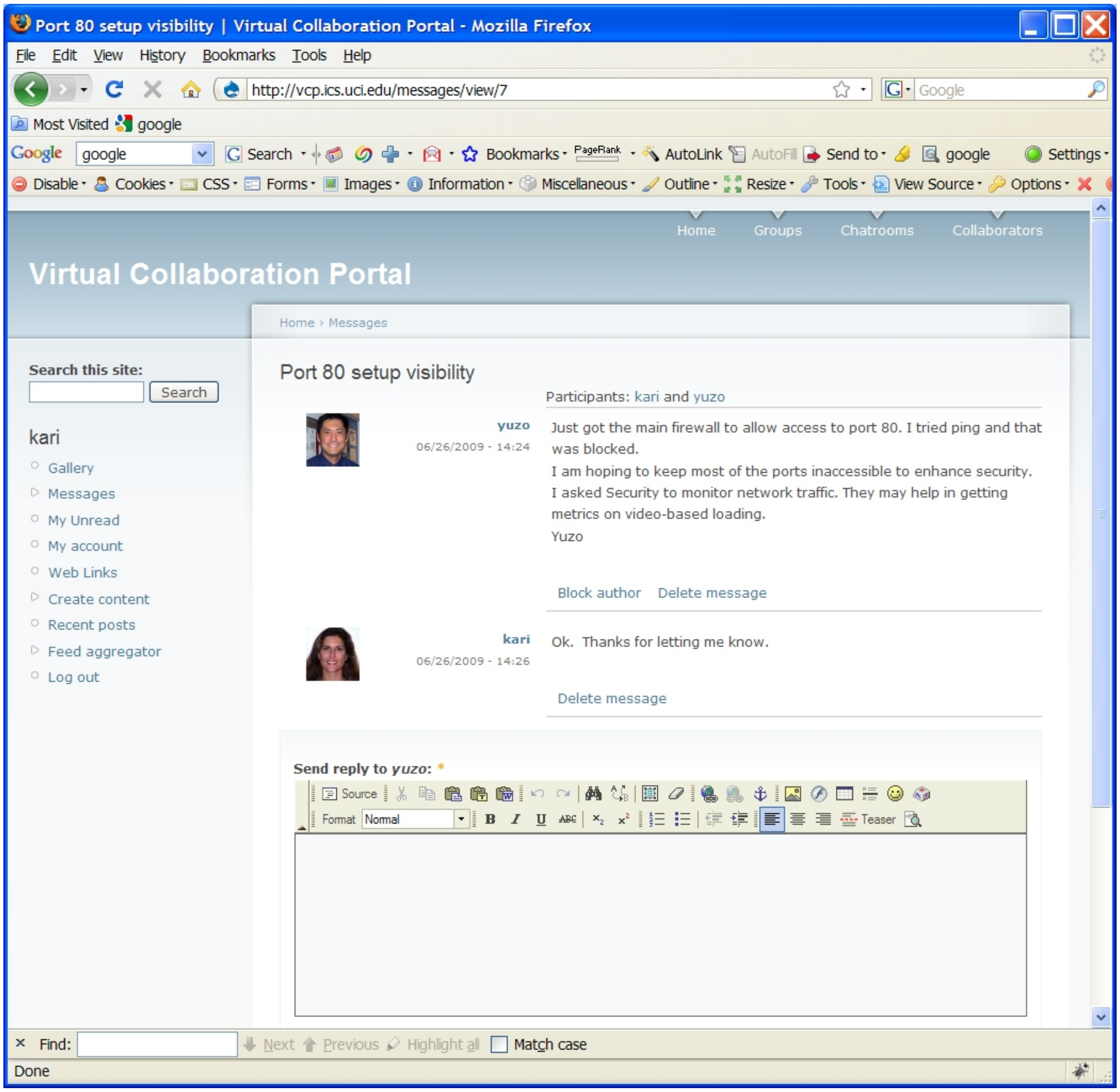


Figure 20: Private message

In addition users can create and/or join chat rooms. A listing of available chat rooms can be viewed by selecting “Chatrooms” in the primary menu in the upper right corner of each page. Figure 21 shows a listing of chat rooms. Within a chat room such as the one shown in Figure4 22 a collaborator can join an existing chat or create a new one. A live chat is shown in Figure 23.

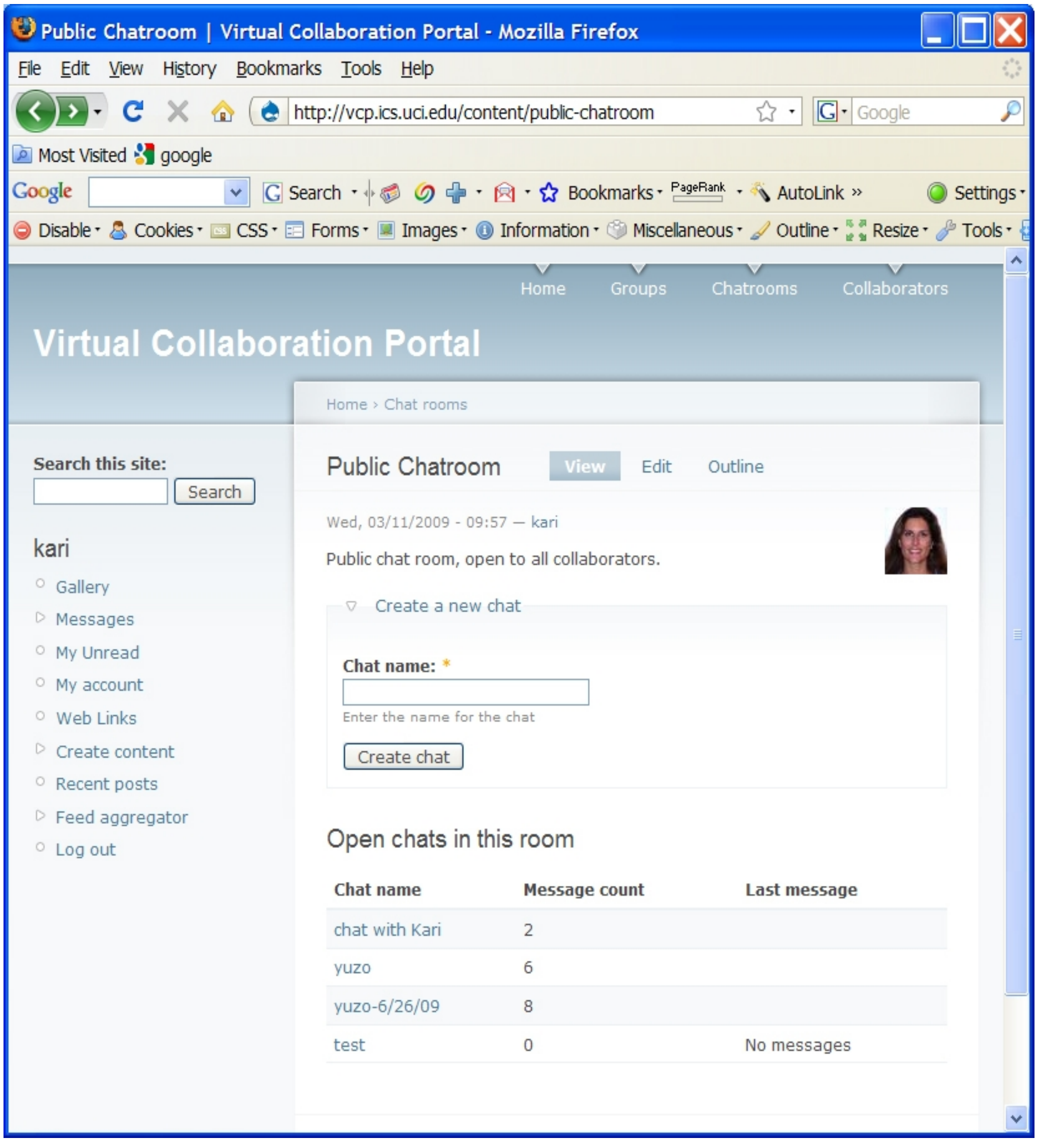


Figure 21: Chat room

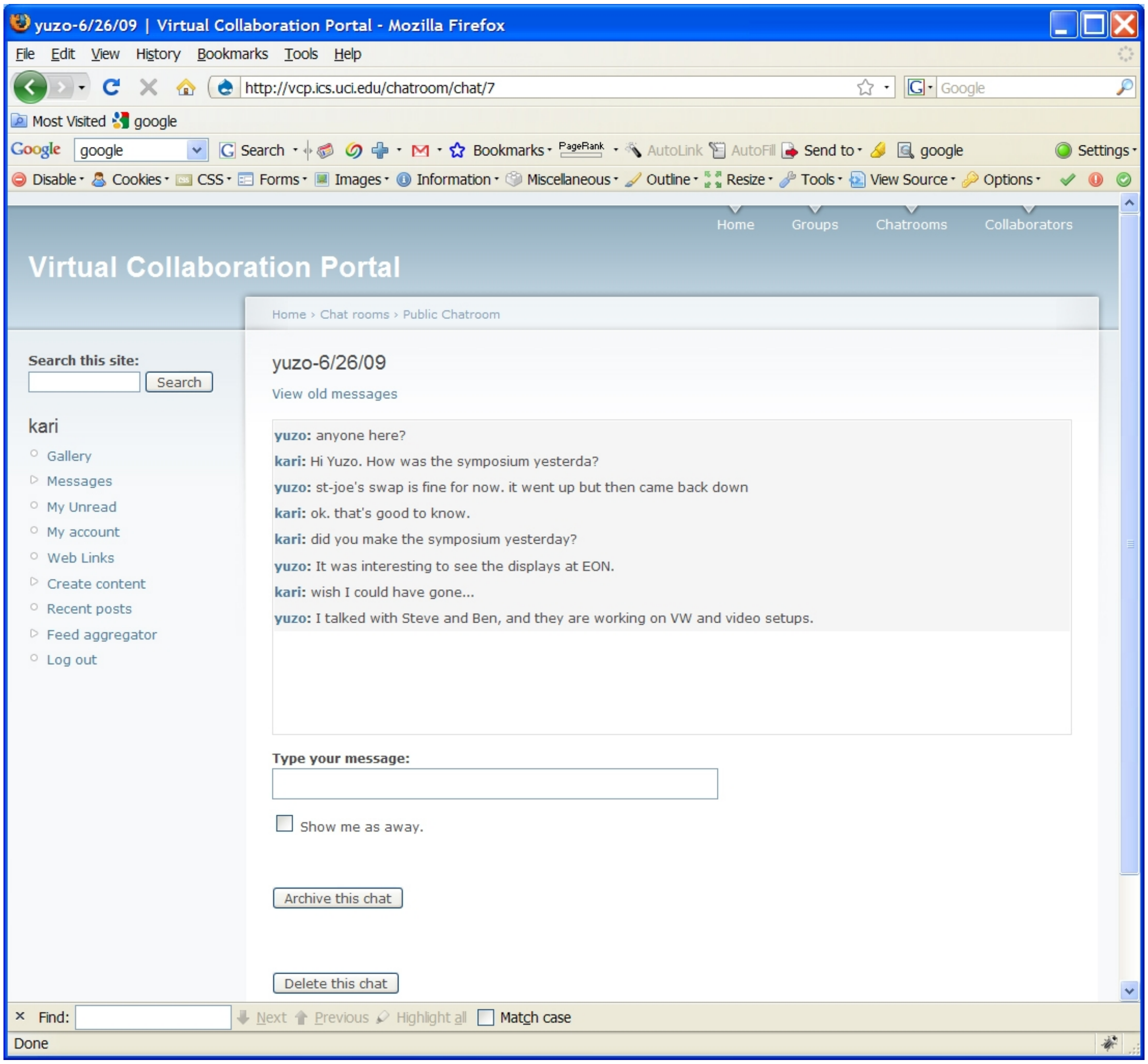


Figure 22: VCP Chat

Web Links

Collaborators can share URLs using Web Links. Any collaborator can create a new group of links or add a link to an existing group. Note that Web Link groups are not related to collaboration groups. Currently there is no collaboration group ownership of Web Link groups. This is an issue for future work. Figure 24 shows an example of web links.

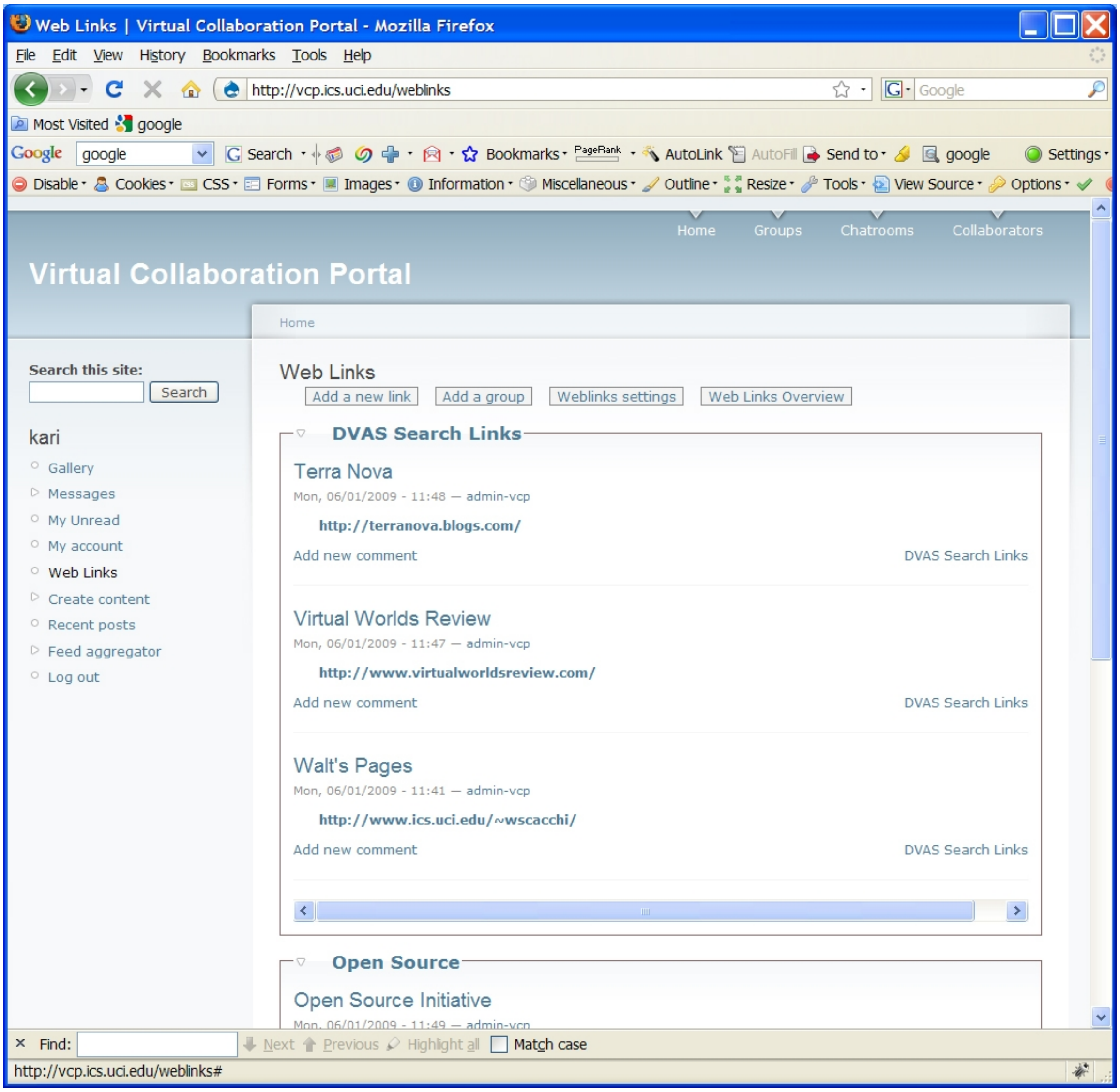


Figure 23: Web Links

Gallery

In addition to being able to post video content, we needed to have the ability to collect and catalogue videos in a centralized location. To this effect we've added an integration of Gallery2, a popular, full featured web-based photo album with support for video. The gallery is accessible by selecting "Gallery" from the navigation menu in the left column. Each collaborator can create his or her own album. The VCP administrator can also create an arbitrary number of additional albums. Permissions for viewing, adding, editing and commenting can be set for 'administrator', 'registered collaborators', or 'everyone'. We have been using the gallery to catalog videos as part of our research. The gallery is depicted in Figure 24.

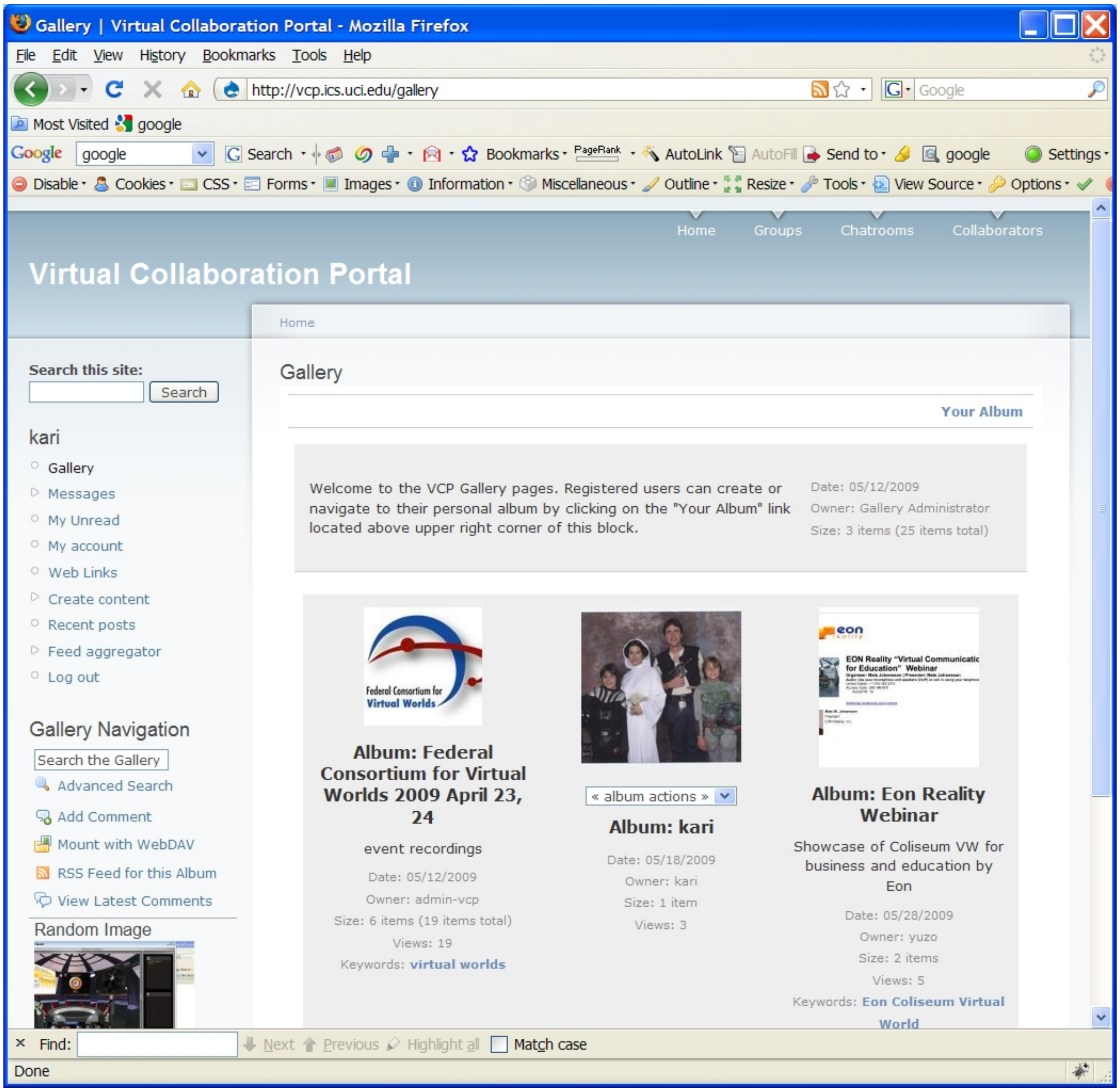


Figure 24: Gallery 2 integration

Figure 25 shows an album of videos that were captured from webcasts of the *2009 Federal Consortium for Virtual Worlds* (see http://www.ndu.edu/irmc/fedconsortium_post_program.html) for further information.

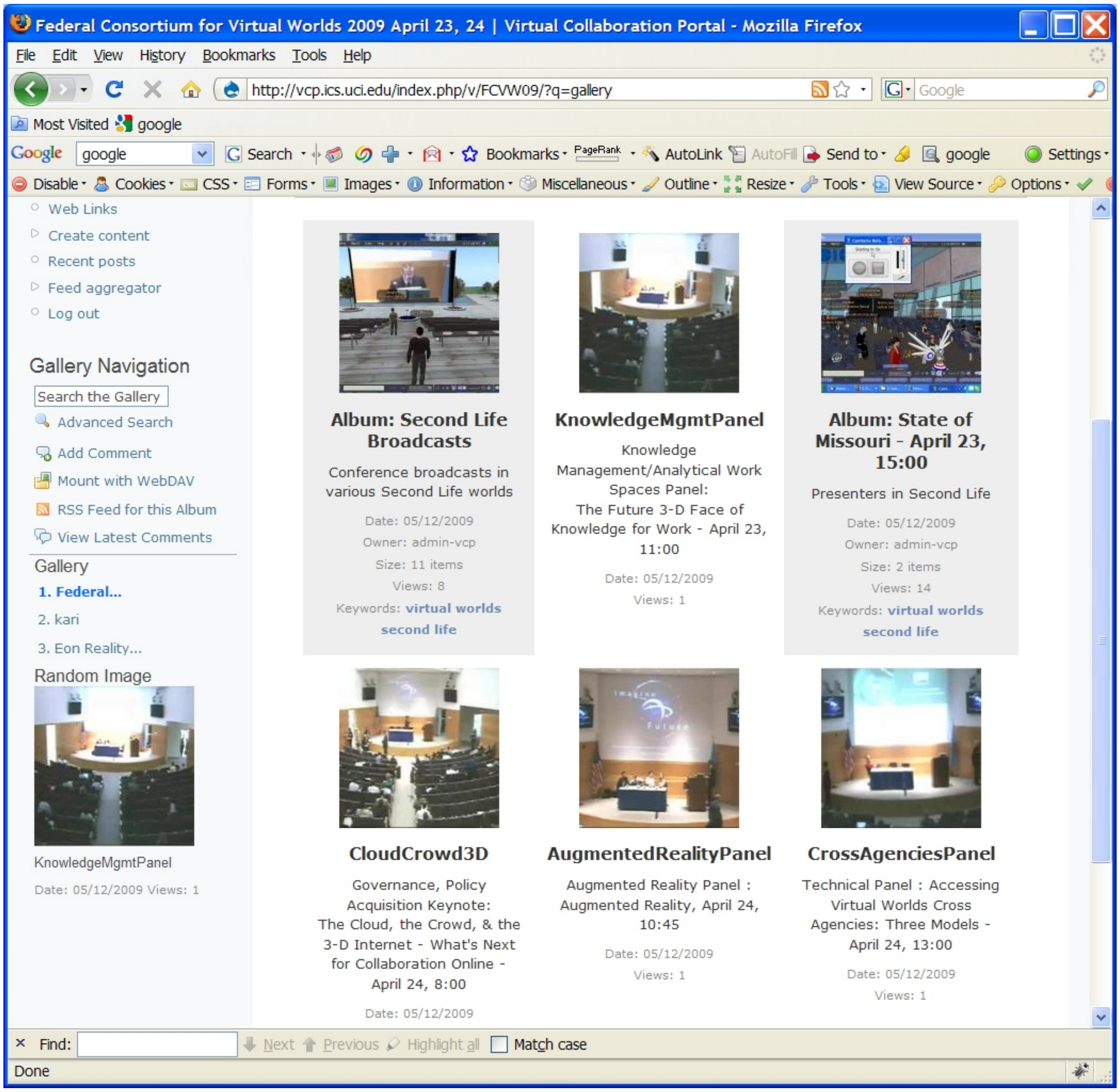


Figure 25: Video album for FCVW'09

And Figure 26 shows video playback of a Second Life capture from within the gallery. In this video a group of rough avatars have taken over an on-line session of the FCVW.

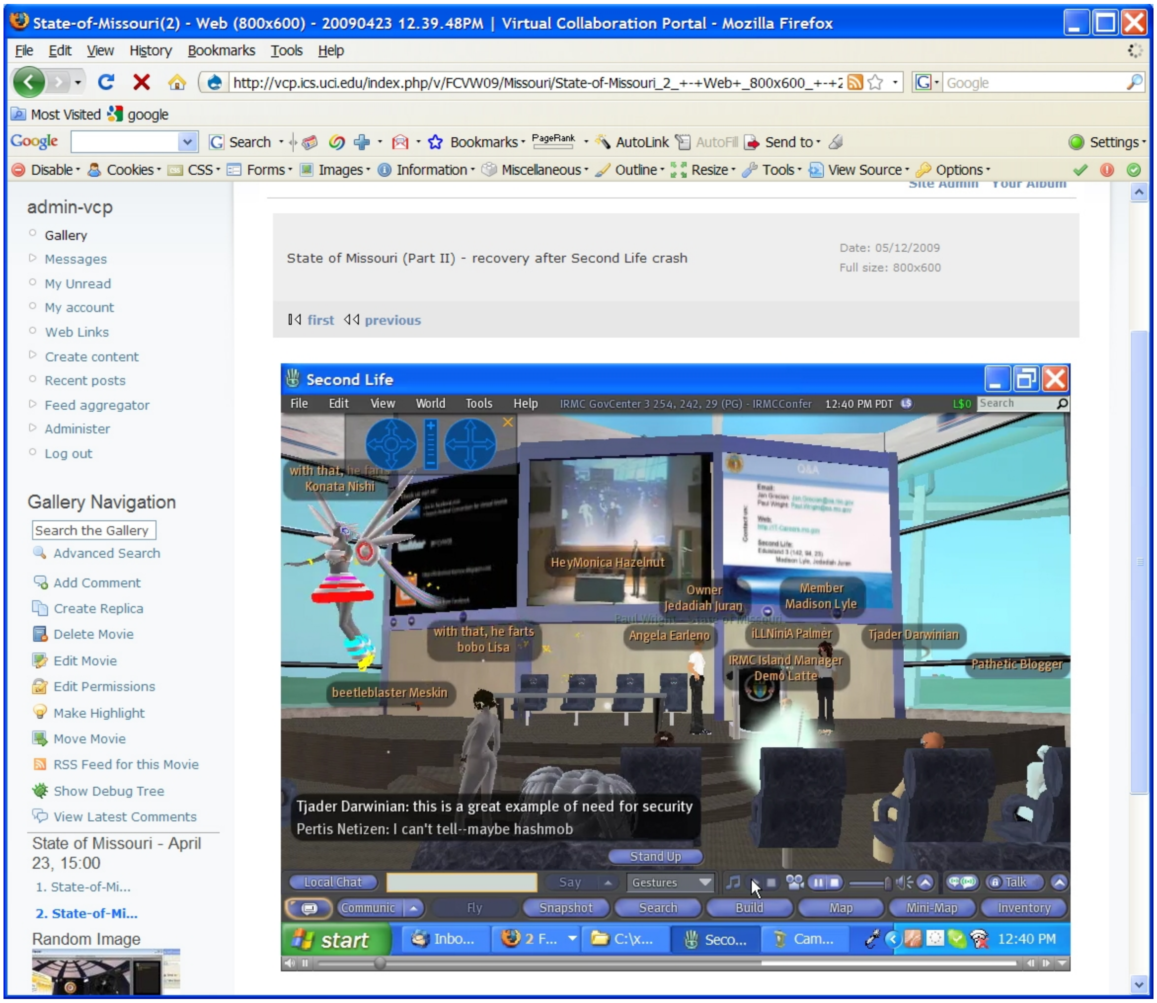


Figure 26: Video playback in Gallery

Search

VCP includes an integration with a Google Custom Search Engine (CSE). A custom search engine was created in a separate Google account. A list of URLs were specified to which searches will be restricted. The CSE is shown in Figure 27.

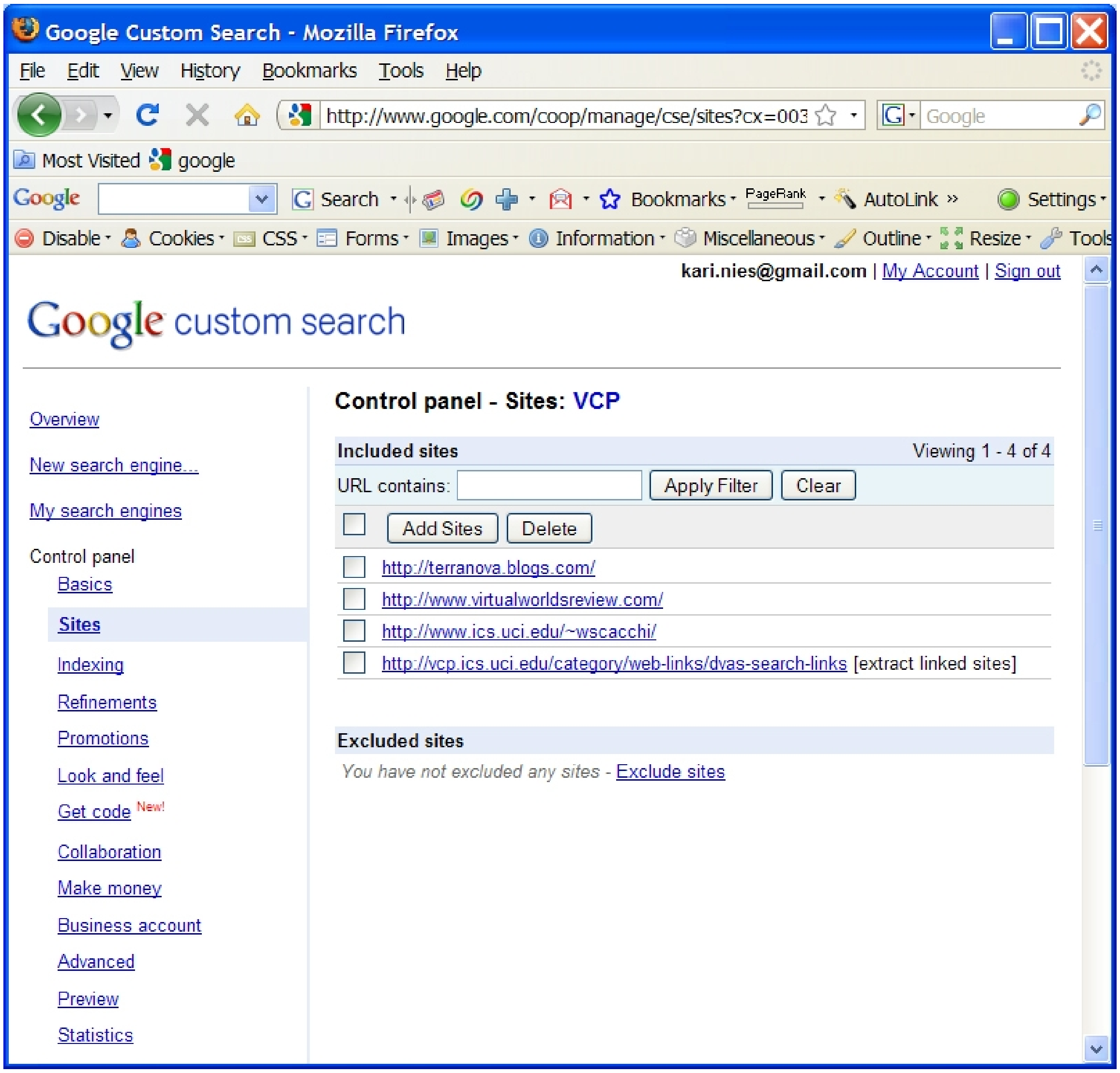


Figure 27: Google Custom Search Engine

A special VSP search option uses the custom CSE and displays the results from within the VSP website. An example search is shown in Figure 28.

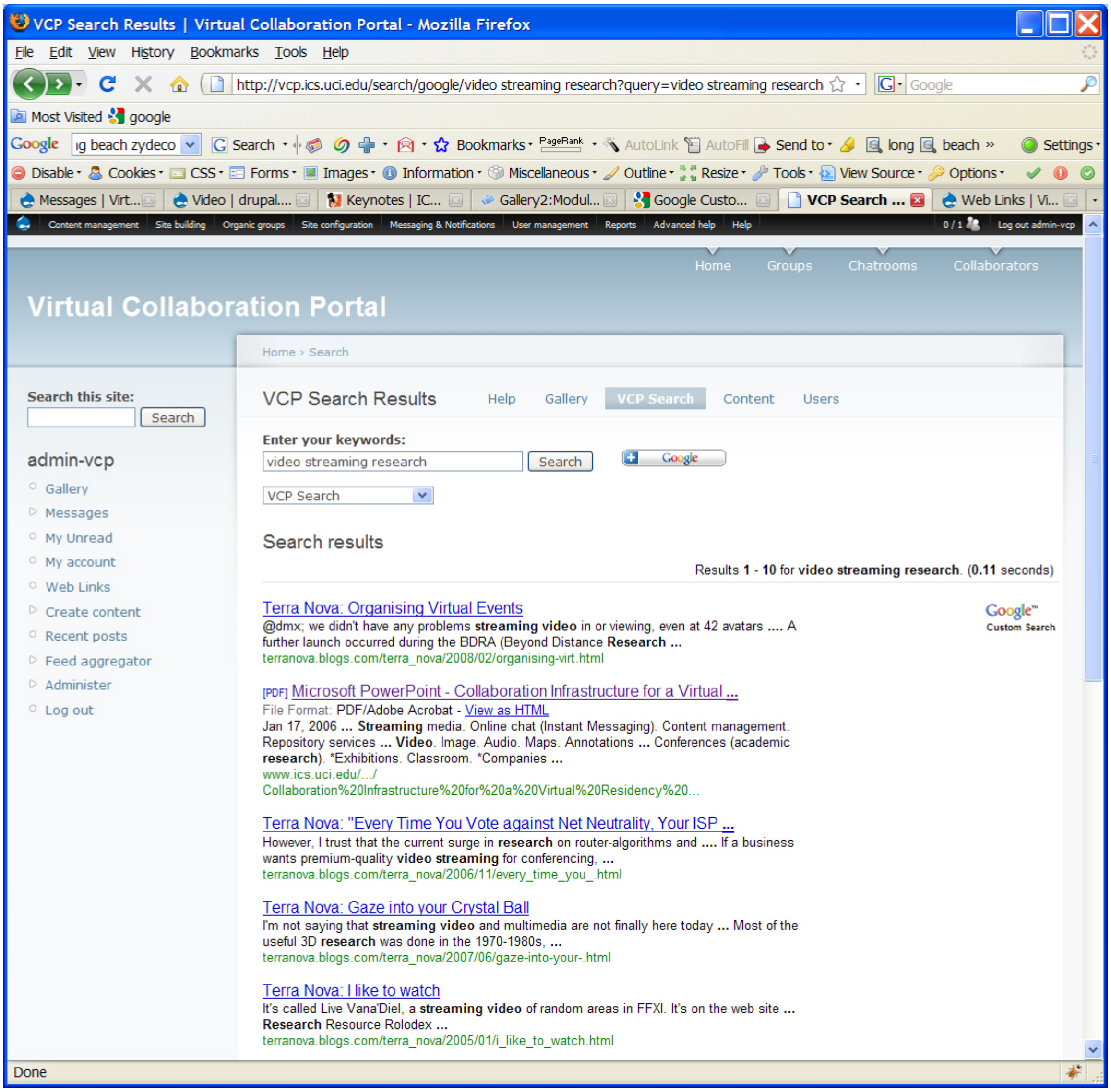


Figure 28: VSP CSE search results

Future Work

Below we discuss some areas for future work.

Search

The current CSE integration uses a Drupal module that only supports a single CSE. This module will need to be modified to support multiple instances and those instances should be enabled with group permissions. In addition, rather than configuring the search sites through a Google account, it would be preferable to pass the sites to the search engine via the CSE API. These sites should be taken from a designated Web Links group. As part of this implementation Web Links will also need to be modified to be group-enabled such that different collaboration groups can create and manage their own Web Links groups.

Co-Authoring

We would like to develop better support for co-authoring on-line documents and are looking in to add wiki type support for easily creating and linking co-authored pages.

Video Conferencing

Unlike the original VCP, we currently do not have any support for video conferencing. Currently Drupal does not provide any suitable modules for this. We are monitoring the Drupal media offerings as they emerge for Drupal version 6.0. We may have to develop our own module for this capability

Additional Video Support

We will continue to seek new ways of capturing, cataloguing, and annotating video.

Integration with Virtual Worlds

We are interested in using the VCP to explore integrations between CMS system and OpenSim.

Group Theming Support

It would be desirable to allow groups to design their own unique home page layout and theming.

VCP Users

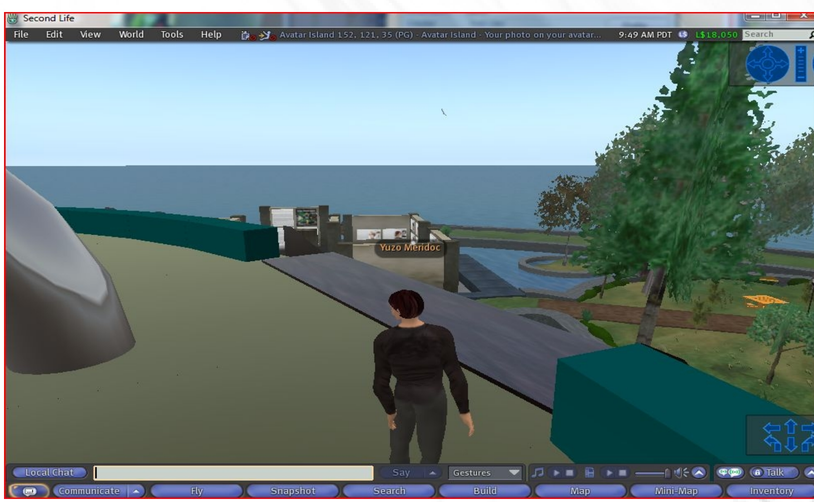
We will be inviting research groups at UCI to begin using the portal in support of better collaboration for testing and feedback.

Virtual World Survey

SecondLife (proprietary), OpenSim, RealXtend

- All in the SecondLife family/code-base
- What platforms and services
- How large/active a community
- What attractive technologies
- How to integrate media (flv, swf, etc.)
- How to integrate sharing (browser, vnc, etc.)

SecondLife



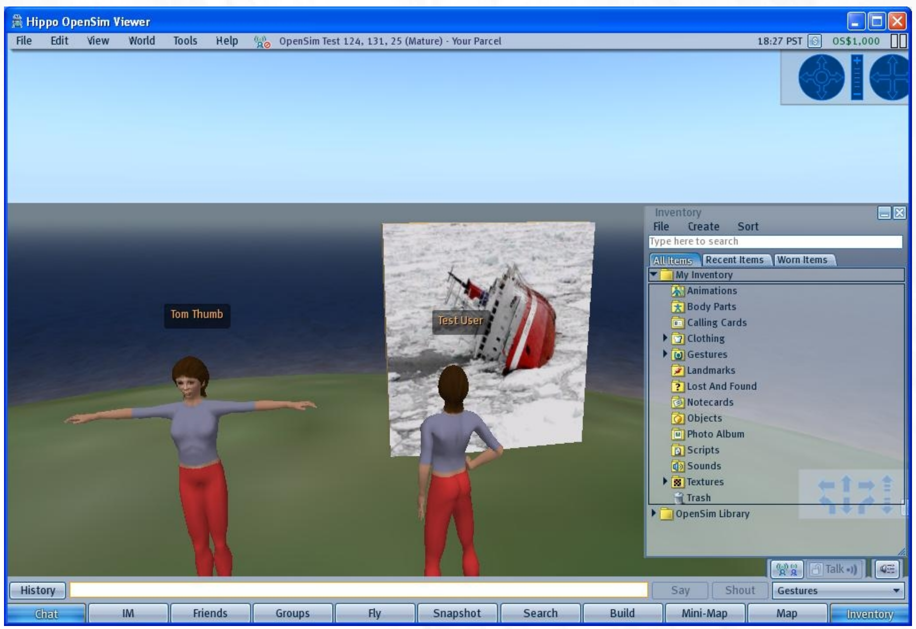
- Very Large User Community (1,000's +)
- Many services/vendors
- Internal and pay html browsers
- Client-side Scripting
  - Linden Scripting Language (LSL)
- Linden hosted service/servers
- Highly integrated Viewer
  - Multiple OS

OpenSim

OpenSim via SL viewer on Mac

- Multiple Viewer support
  - SL, Hippo, etc
- Server based on SL
  - Runs on 32bit Windows and Linux
- ModRex in development
  - RealXtend viewer integration with OpenSim
- BSD license

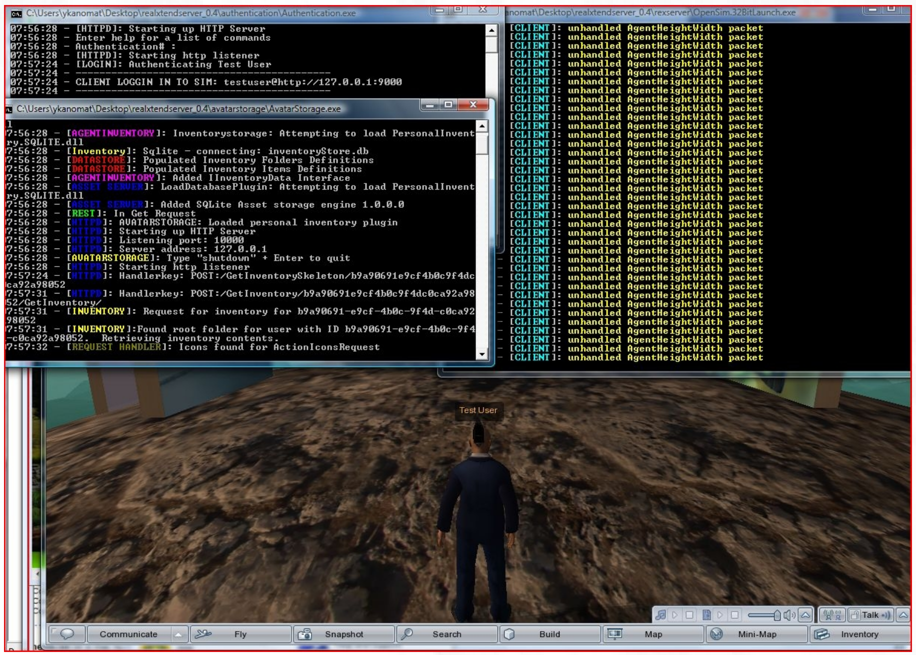
OpenSim



OpenSim via Hippo
viewer on WinXP

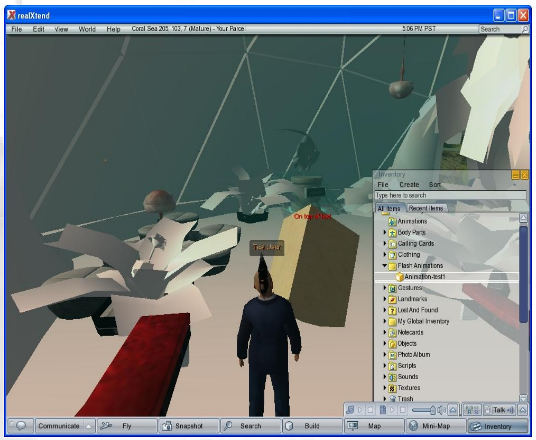
- Large Developer Community (100's to 1,000's)
- Scripting
  - Client-side LSL
- Collab Tools
  - HTML browser as surface texture
  - Skype

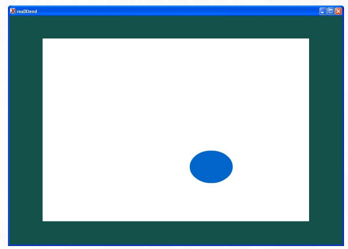
RealXtend Server



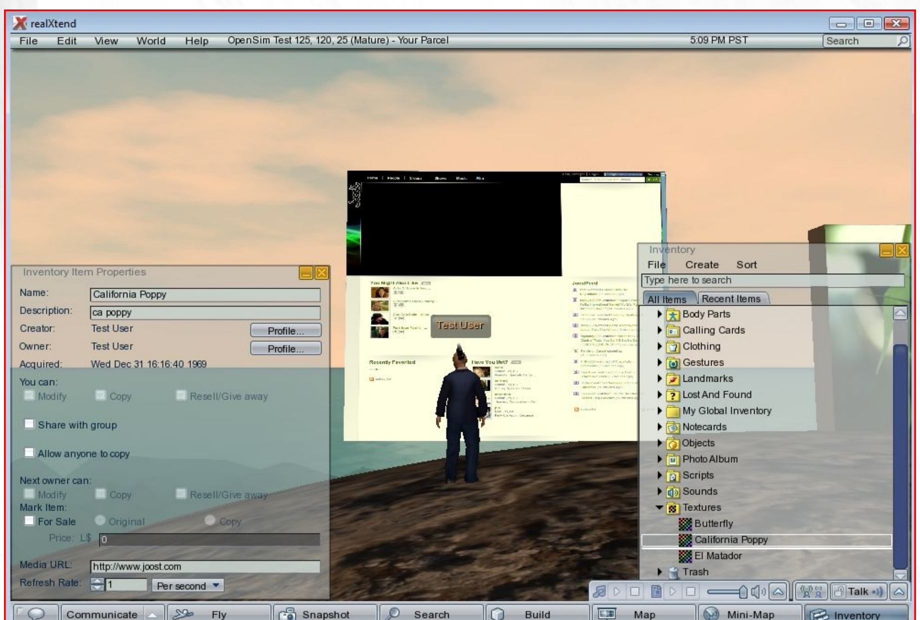
- Based on OpenSim
  - BSD License
- Small Dev Team (6) and Community (~200)
- Scripting
  - Server-side IronPython (allows swf and other media obj.)
  - Client-side LSL
- Windows centered Development

RealXtend viewer 0.4



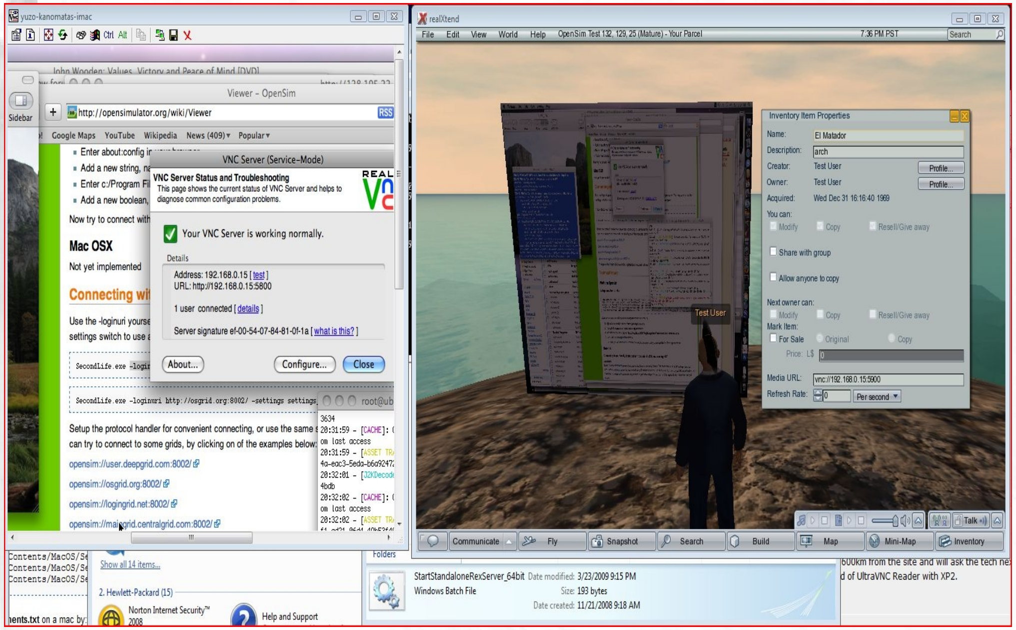


Touch action starts anim.



- Highly Integrated with Server
  - GPL License
  - Planned Mac and Linux versions (0.45 or 0.5)
- High Graphics Req.
  - 3D Ogre Mesh
- Viewer additions
  - Flash Animations
  - Access to VNC
- Client-side LSL Scripting

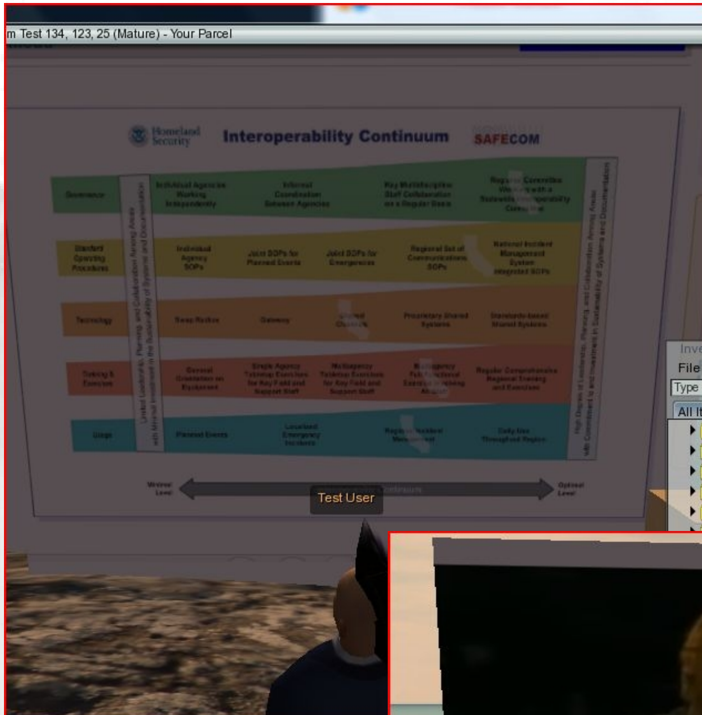
RealXtend viewer Collab

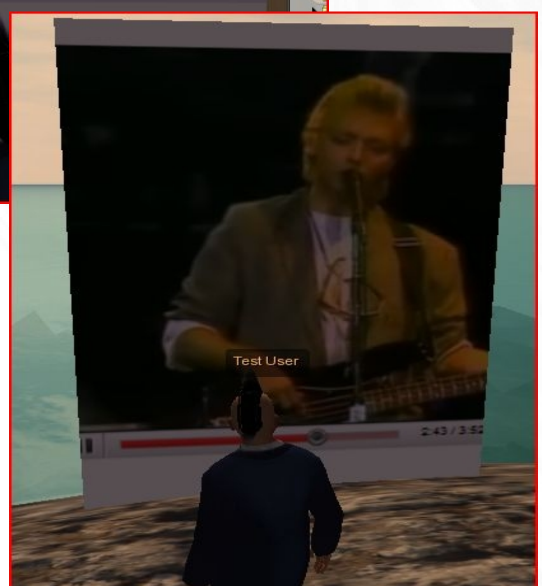


VNC client window on left
VNC projection on a surface in-world on right

- Can share desktop via VNC connection
  - Uses embedded TightVNC client
  - Can connect to TightVNC or RealVNC Server
- User of VNC Client controls displayed items
- Direct support for Skype

RealXtend viewer Collab (in-world browser)





Slides as Flash via slideshare above YouTube video on right

- Can apply browser as a Texture to a prim
  - Slide presentation mashup via slideshare.net
  - YouTube video work-around
- Co-browsing capable via external website using php

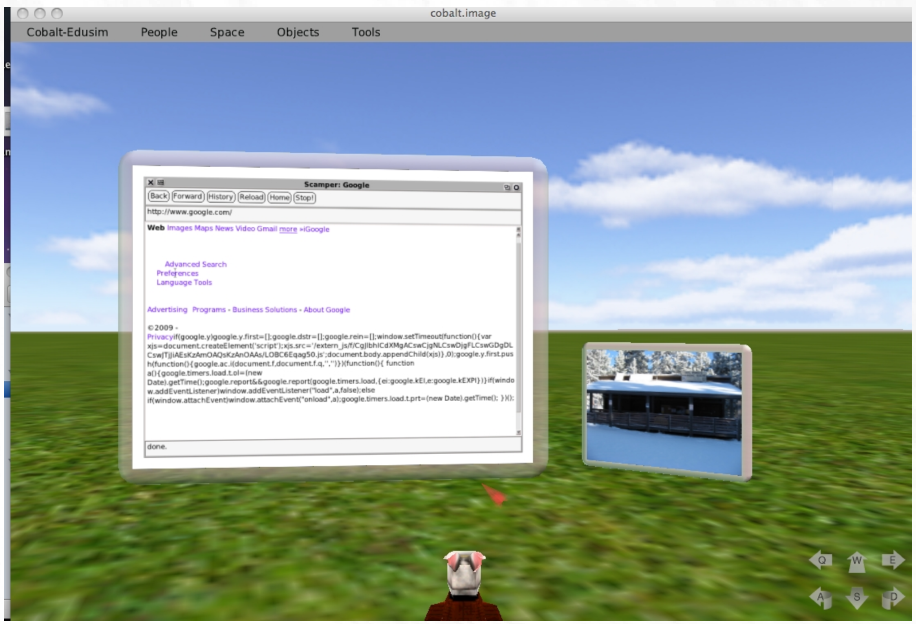
General Comparison

	Community	Collab Tools	Rendering	Add-ons/ Scripting	Viewer/Server Integration
SecondLife	Very Large	Commercial+free	High	Large, but most are commercial	High
OpenSim	Large	Lots of FOSS	Moderate	Large, most FOSS	Moderate
RealXtend	Small	OpenSim's base + additions	Highest	Open Sim's base + Server-side Python	High

Other Possible Candidate

- EduSim (part of Cobalt/Croquet family – Alan Kay)
  - Import of 3D models
    - Some Google 3D warehouse items
  - Included Tools
    - VNC application sharing
    - Limited in-world HTML browser
  - BSD license
- Note: SL and OpenSim may become interoperable, ModRex may become a plugin to OpenSim

EduSim



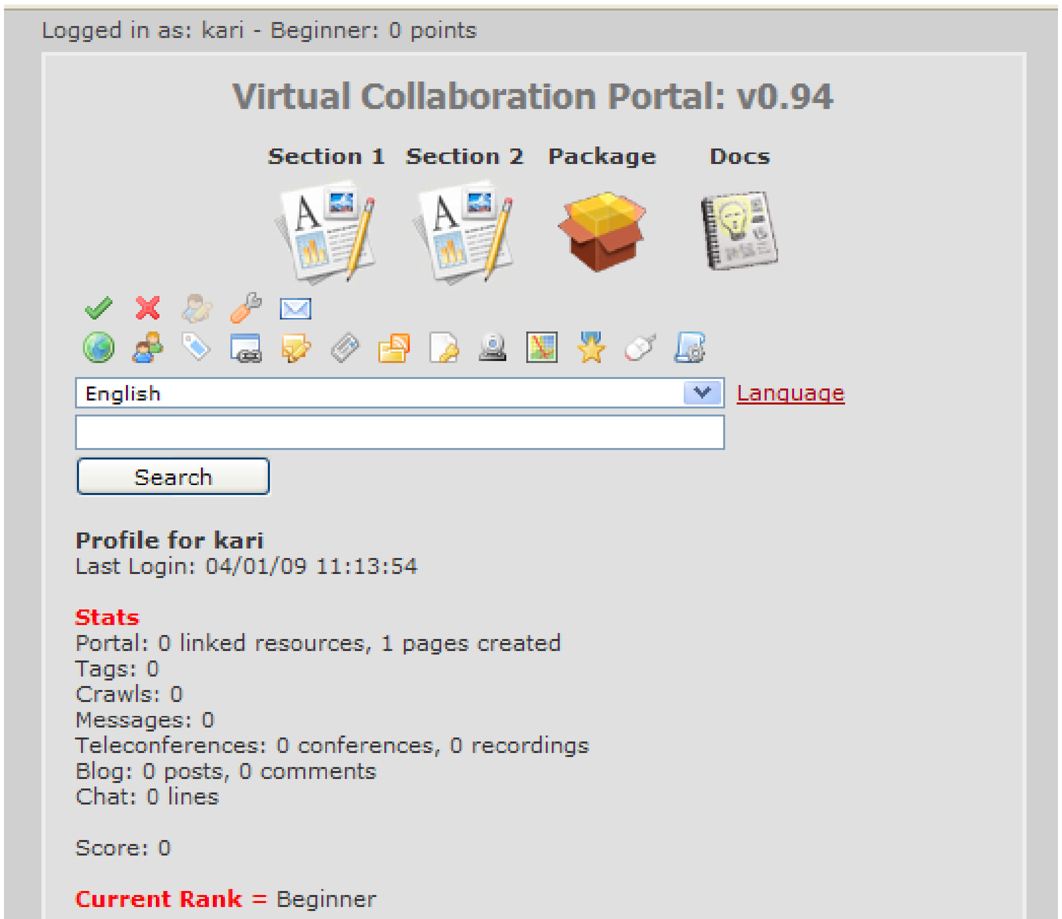
Google homepage on left
Jpeg on right

- Multiple platforms
  - Squeak-based (SmallTalk)
  - Low graphics req.
- Understood objects
  - Some HTML and media
  - No Flash

Next Generation Virtual Collaboration Portal

An Integrated OSS CMS-Based Solution

VCP Features



Portal CMS

- User Login/Profile
- User Preferences
- Members List
- Messages
- Groups
- Tags
- Linker
- Rank
- Content Manager
- Templates

Tools

- Wiki - Mediawiki
- Blog - B2Evolution
- News Feeder - Gregarious
- Version Control System - DVCS
- Teleconference Suite - Flash/PHP Chat
- Automated Caller - Asterisk
- Mapping Utility - Google Maps
- Search Engine - Lucene/Nutch
- Web Crawler - Nutch

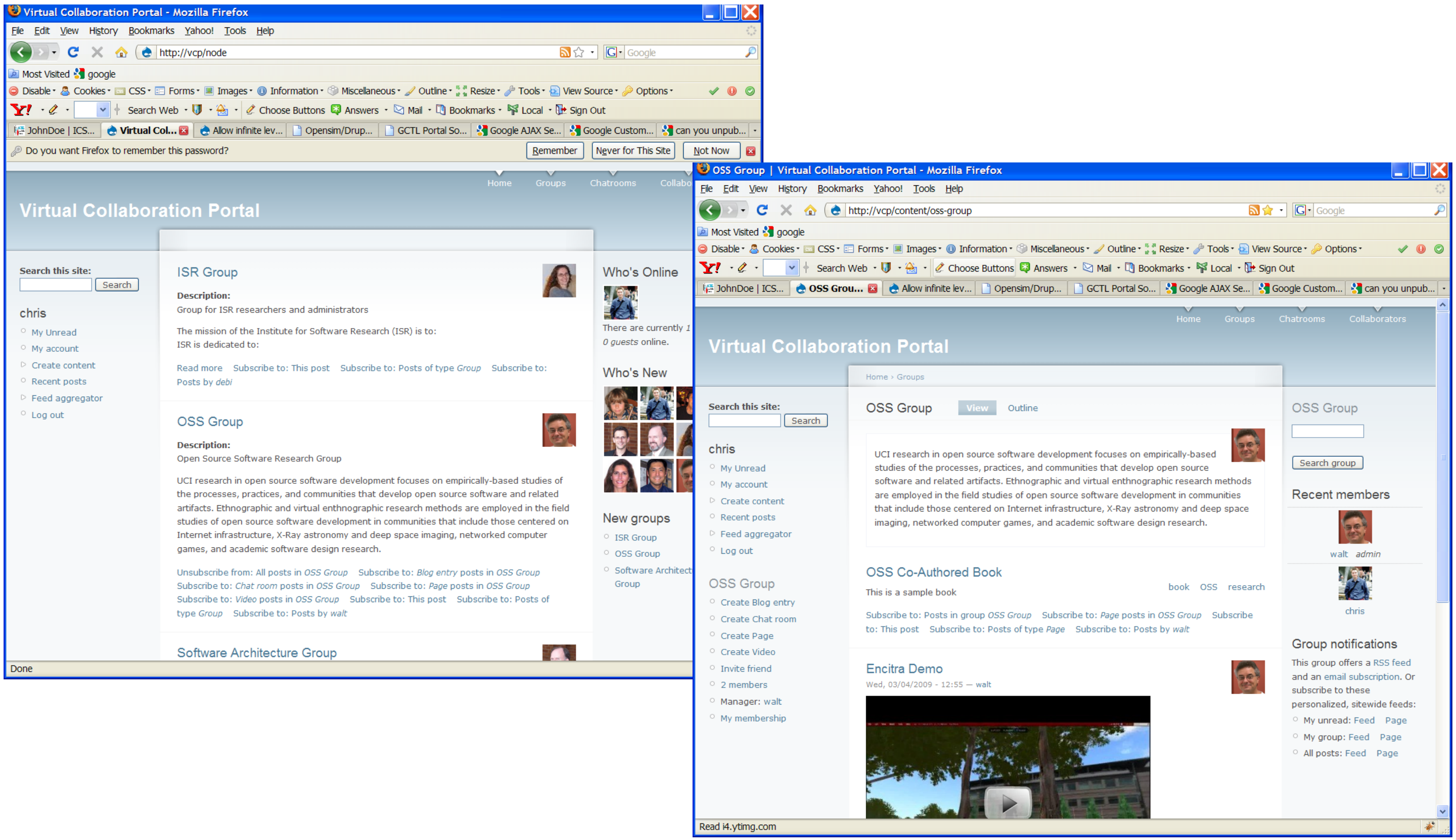
Open Source CMS Motivation

- Current Implementation
  - Home grown CMS using PHP and MySQL
  - Brittle due to API and structural evolution of a myriad of integrated external tools.
- Leverage off an Open Source CMS with many needed features built in.
- Drupal
  - Content management framework geared towards configurability and customization.
  - Very large community (> 450K user accounts, > 2K developers)
  - Thousands of user contributed modules (> 3700)
  - Top 3 in market share (along with Wordpress, Joomla)

Drupal Module Evaluation

Login/Profile, User Preferences, Members List, Content Manager, Templates, Blogs, Linker, Search	Drupal Core
Groups, Tags, News feeder	Quality modules
Messages	Quality modules for subscription-based notifications. So far, no direct user to user messaging.
Rank	Instrument existing modules to tally usage. Implement Rank module using 'roles'.
Wiki	No wiki module. However wiki functionality can be built on top of existing content types (usually Books) using wikitools, filters, diff, freelinking. wysiwyg editor option.
Version Control (DVS) – upload, checkout, edit	File uploads supported in core. Version control api with functional backends for CVS, SVN, Git, and Mercurial. No core DMS. One module integrating KnowledgeTree.
Teleconference Suite	Primitive module for chat. Drupal 6 Media sprint in progress. Better media management. Will include php stream wrappers for 3rd party multimedia repositories, may allow integration with tools such as DimDim.
Web Crawler	Crawler to scan key sites and gather new content. Use Drupal cron jobs to run periodically. Google Custom Search API.
Mapping Utility	New GMap module released Mar'09.
Automated Caller	No modules. Might have to create a new Astericks wrapper.

Drupal-based VCP Prototype



OpenSim Integration

- Attempt to integrate a CMS with a 3D World.
- Access content in response to OpenSim events
- Drupal as CMS
  - Active Drupal Group on SecondLife integration.
    - http://groups.drupal.org/secondlife
  - Second Life Framework Module
    - http://drupal.org/project/secondlife

Understanding and Supporting Distributed Team Work: Position Paper

Walt Scacchi
Institute for Software Research
University of California, Irvine
Irvine, CA 92697-3425 USA
http://www.ics.uci.edu/~wscacchi
wscacchi@ics.uci.edu

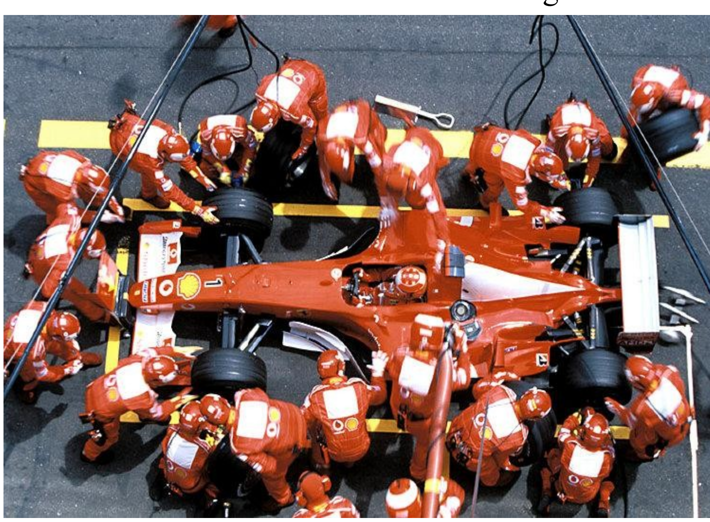
Introduction

I have been engaged in empirical studies of software development teamwork for the past 25 or so years [1-10]. Along the way I have examined different kinds of software development activities, processes, work practices, projects, and communities both within and beyond the boundaries of a single enterprise. This includes a number of efforts where I organized and directed the development of large-scale software systems with teams totaling up to 80 or more developers. Most recently, I have been engaged in studies of free/open source software development (FOSSD) in a variety of domains including astrophysics, bioinformatics, higher education computing, networked computer games, and military computing, as well as mainstream FOSSD projects and project communities [3,4, 8-10]. Study of FOSSD is noteworthy here since the most FOSSD teamwork occurs in decentralized settings where developers work in information spaces with online artifacts that are network-centric and logically collocated, rather than in places where they are physically collocated [cf. 7]. I have examined a variety of situations where cooperation, coordination, and collaboration where central issues under study, and this background serves to inform my position for how to support distributed team work.

Accordingly, I want to present my position in terms of the following topical framings. First, I characterize some of what I know about radically collocated teamwork. Second, I seek to provide results from a study of collocated software development teamwork to help inform our understanding of what kinds of concerns may need to be addressed to more effectively support distributed computer supported teamwork. Last, I want to posit some ideas and opinions about emerging forms, situations, and venues for supporting distributed teamwork. Each follows in turn.

Supporting Radically Collocated Team Work

While there is real interest in understanding how best to support distributed teamwork when the work is non-routine and performed by scientists, engineers, technical specialists or other high-value workers, it is also useful to first look at radically collocated teamwork which is routine and well-structured. This may help identify where issues of coordination arise, and how work can be organized and performed so as to maximize coordination effectiveness and efficiency, while minimizing conflicts. Consider the adjacent exhibit, a photograph of more than 20 technical specialists working as a team to service a modern Formula 1 race car in a strongly collocated and extremely coordinated manner. (An explanation of the team roles and activities can be found at http://www.f1technical.net/articles/3337, and a video example



can be seen elsewhere[1]). The work activities are directed via a race team's integrated multi-display IT command and control system (not shown) linking on-vehicle telemetry data sensors, digital voice communications, and remote data processing facilities. The observable teamwork is (a) performed in very short duration time-boxes (productivity is measured in seconds elapsed), (b) choreographed and well-rehearsed in advance, (c) highly specialized into narrow functional-skill roles, (d) synchronized but generally not collaborative, and (e) pre-positioned in marked areas with specific resources/tools, among other things. Mistakes or miscues in teamwork [6] or unanticipated maintenance work [1] can have substantial costs or severe consequences. The F1 pit stop is more similar to a ballet than to team sports like basketball or soccer where collaboration (or collaborative play) is paramount. So items (a) through (e) point to possible ways to improve the productivity of teamwork that can be radically collocated when high levels of technical specialization and expertise are involved, and when focused, nuanced performance is needed.

Supporting Software Development Team Work

Software development teamwork is often organized around system architectural elements like sub-systems which may then be assigned to small teams of 3-8 developers for development. In one study [2], I had five teams of 5-7 developers assigned to produce and deliver a set of operational software specifications and associated documents over a two week period. All teams were to follow the same overall process, which included collaborative activities and the use of advanced software development tools (e.g., software specification analyzer). Each collaborative work meeting of each team was observed, and recurring social interactions (e.g., resource arrangements, division of labor and expertise, reviews of work in progress) were notated and coded for analysis. However, team members could also work outside of meetings. The following exhibit presents summarizes the codings of work structures each team enacted through the process. A few key observations: (a) the teams naturally structured themselves into one of six different configuration patterns or "work structures[2]", (b) teams planned for shifts in their work structures, (c) some teams experienced unplanned/unanticipated shifts in their work structures. The results are ordered left-to-right by quality of results delivered (hi-low, measured by automated tools) and also from right-to-left by productivity (hi-low, measured by total time on task). However, there is only a coincidental relationship between quality and productivity. Instead, we found that (d) whether a team followed its work structure plan or not did not determine overall quality or productivity, but (e) teams that were more collaborative produced higher quality results, while those that were more dispersed and relied on explicit coordination were more productive. As such, our conclusion was that computer supported teamwork environments need to explicitly support alternative teamwork structures and parameters to be most effective, or else such environment may only be effective or efficient some

Team ID	T1	T2	T3	T4	T5
Team Size	6	7	7	7	5
Reusable Exemplar	no	yes	yes	yes	yes
PROCESS					
A. Pre-planning task	N->R->I	N->R->I	N->R->I	N->R->I	N->R->I
B. Planning task	N	N	N	N	N
c.	I	I	I	I -> S	I -> S
d.	I	I+	I	S	S
e.	P (D,I,I)	P (D,I,I)	P (D,I,I)	P (D,S->I,I)	P (D,D,I)
f.	D	D	D	D	D
g.	R	R	R	D	D
h.	D	I	I	D	D
C. Develop preliminary	I -->>	I+	I+	I+ -> S+	I -> S+
(informal) specification	N -->>				
	R -->> I				
D. Develop formal	I+	I+	I+	S+ -->>	S+
(processable) spec.				N -->>	
E. Document	N -->> D	P+	P+	N -->>	P(D,D,I)
write-up				P(D,S->I,D)	
F. Documentation	D -->>	D+	D+	D+	D+
integration	N -->> I				
G. Document review	R	R	R	N -->> R	D
H. Prepare for	D -->>	I+	N -->> I+	N -->> I+	D+
Delivery	N -->> I				

1 See http://www.youtube.com/watch?v=TQoGCq1caI4&feature=related

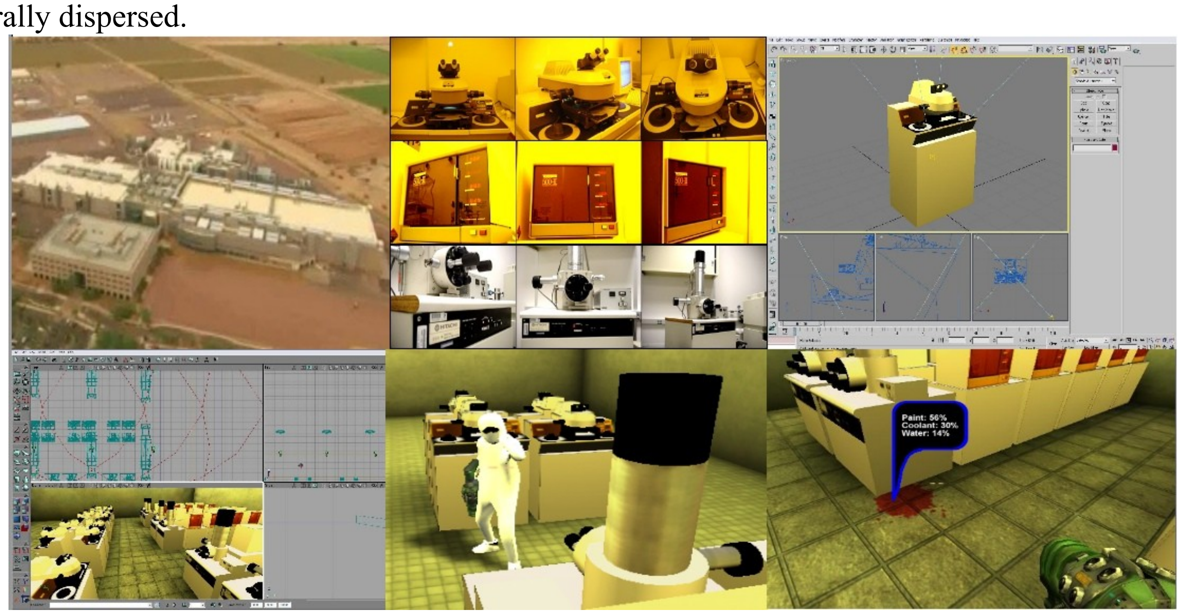
2 We identified these as **N**egotiated, **I**ntegrated, **R**eplicated, **D**elegated, **P**rediscrimated, or **S**eparated. They are ordered from most collaborative to most isolated (and thus dependent on explicit coordination).

of the time, and be obstructive at others times. (This might explain why some teams find certain tools effective to support their work, while other people have trouble using the same tools—support environments need to complement shifting work structures, rather than reinforce a predetermined one).

Looking Forward

From here, I am also engaged in studies looking into the development and deployment of distributed teamwork environments that employ virtual/mixed reality interfaces or networked computer game play [12]. These efforts may serve to inform discussion about emerging concepts and technologies for supporting distributed or decentralized teamwork support. For example, in one effort, I am looking into at emerging work practices associated with teams that are collocated in meeting rooms that are physically separated, but share a common virtual video wall for interaction with online objects and videoconferencing, as suggested by the image below.

In another project, I am leading an effort to develop game-based virtual work environments that can be used to create teamwork simulators for activities associated with semiconductor or nanotechnology fabrication facilities. In such settings, virtual collocation of teamworkers, especially new, untrained workers, is perceived as less costly and potentially more effective than physical collocation with actual fabrication facilities. Similarly, virtual collocation in a simulator also offers the potential for remote diagnosis of fabrication problems or breakdowns that may arise in facilities that are geographically and culturally dispersed.



Last, lest we forget that school classrooms or similar venues like regional science centers are places where students learn about working together. These settings increasingly provide radical collocation of distributed teamwork that involves ever more sophisticated, multi-media computing environments. The last exhibit below provides some documentation of a recently deployed distributed teamwork environment at a regional science center where teams include family members and ad hoc student groups who find that working collaboratively with/through online collaborative (learning) work environments can be a lot of fun. More details can be found elsewhere on this effort [11].

Acknowledgements: Research support from NSF grants #0534771 and #0808783, Naval Postgraduate School, and Intel. No endorsement implied.

References

1. Bendifallah, S. and Scacchi, W (1987). Understanding Software Maintenance Work, *IEEE Transactions Software Engineering*, SE-13(3), 311-323.
2. Bendifallah, S. and Scacchi, W. (1989). Work Structures and Shifts: An Empirical Analysis of Software Specification Teamwork, *Proc. 11th. Intern. Conf. Software Engineering*, Pittsburgh, PA, ACM Press, 260-270, May.
3. Elliott, M. and Scacchi, W. (2005). Free software development: Cooperation and conflict in a virtual organizational culture. In S Koch, (Ed.), *Free/Open Source Software Development*, Idea Group Publishing, Hershey, PA, 152-173.
4. Jensen, C. and Scacchi, W. (2007). Role migration and advancement processes in OSSD projects: A comparative case study, in *Proc. 29th Intern. Conf. Soft. Eng,* ACM, Minneapolis, MN, 364-374.
5. Kling, R. and Scacchi, W. (1982). The Web of Computing: Computing Technology as Social Organization, in M. Yovits (Ed.), *Advances in Computers*, 21, Academic Press, New York, 3-85.
6. Mi, P. and Scacchi, W. (1991). Modeling Articulation Work in Software Engineering Processes, *Proc. 1st. Intern. Conf. Soft. Processes*, IEEE Computer Society, Redondo Beach, CA,188-201.
7. Noll, J. and Scacchi, W. (1999). Supporting Software Development in Virtual Enterprises, *J. Digital Information*, 1(4).
8. Scacchi, W. (2005). Socio-Technical Interaction Networks in Free/Open Source Software Development Processes, in S.T. Acuña and N. Juristo (eds.), *Software Process Modeling*, 1-27, Springer Science+Business Media Inc., New York.
9. Scacchi, W. (2006). Understanding Free/Open Source Software Evolution, in N.H. Madhavji, J.F. Ramil and D. Perry (Eds.), *Software Evolution and Feedback: Theory and Practice*, John Wiley and Sons Inc, New York, 181-206.
10. Scacchi, W. (2007). Free/Open Source Software Development: Recent Research Results and Methods, in M. Zelkowitz (Ed.), *Advances in Computers*, 69, 243-295.
11. Scacchi, W., Nideffer, R., and Adams, J. (2008). Collaborative Game Environments for Informal Science Education, *Proc. 2008 International Symposium on Collaborative Technologies and Systems*, Irvine, CA, May.
12. Scacchi, W., Nardi, B, *et al*. (2008). Decentralized Virtual Activities and Technologies: A Socio-Technical Approach.

Other topics in game culture and technology research and practice at UCI

unexceptional.net

Robert F. Nideffer
University of California, Irvine
Irvine, California 92697,
USA
+1 949 824 4218
nideffer@uci.edu

Alf Inge Wang
Norwegian Univ. of Sci & Tech/
University of California, Irvine
Irvine, California 92697, USA
+1 949 824 9495
alfw@idi.ntnu.no

Alex Szeto
University of California, Irvine
Irvine, California 92697,
USA
+1 949 824 4218
alex.szeto@gmail.com

ABSTRACT

This paper describes the pervasive game unexceptional.net. Unexceptional.net is a unique pervasive game that is story-driven where the story evolves through multiple media channels interacting with the player. The game interacts with the user on a cellphone and a PC, and the game provides unique gaming experience by introducing gameplay elements through comics, web hacks, Blogs, a 3D-client, a database transaction visualizer, a cellphone client and an Applet client. The paper describes the process of building this game through integrating various open source software and running multiple student projects. Further, the paper reveals challenges in building multi-channel pervasive games and identifies how this was solved in the unexceptional.net project. The contribution of this paper is the description of a novel pervasive game and the experiences from designing and implementing this game.

Keywords

Pervasive Gaming, Game Concept, Software architecture, Game Development.

1. INTRODUCTION

Research within pervasive gaming has had a growing interest the last years and the development of mobile and context-aware computing opens for new opportunities in this domain. Traditional computer games are attractive for people as they draw them into an interactive imaginative world through sound and graphics [1]. Good computer games immerse the player through fantasy, providing the appropriate challenge, and stimulating the player’s curiosity by hinting to the player that something will be reviled if the player succeed to the next level [2]. Games are now becoming more important as a social arena and in games like World of Warcraft the social aspect plays a major role [3]. Pervasive gaming tries remove the boundaries that games are only played on your computer or game console and does not relate to the real world. In pervasive gaming, the games can be played in various settings, on multiple devices, and interaction with real physical things and people. Pervasive games can also be classified into their own genre of games like *smart toys* – electronic toys with sensors that include game mechanics, *affective gaming* – games that influence and/or are influenced by emotions, *augmented tabletop games* – mix traditional tabletop games with computer games, *location-aware games* – games where the gameplay is affected by the players physical location, and *augmented reality games* – games that uses augmented reality to allow gameplay that mixes real (physical) and virtual objects [4].

In this paper we describe the project unexceptional.net, conceived and directed by Professor Robert Nideffer at Department of Studio Art and Informatics, University of California, Irvine, over a 3.5-year period between Fall 2002 and Spring 2006. Unexceptional.net can be classified both as an affective game and a location-aware game according to the taxonomy presented above. The uniqueness of this game can be characterized in that it is story-driven, and that the story evolves for the player through multiple ways of interaction. This paper describes the concept of the game, architectural considerations, and experiences from implementing the game.

The rest of the paper is organized as follows. Section 2 describes an overview of the game. Section 3 describes experiences from designing and implementing the game. Section 4 describes related work, and Section 5 concludes the paper.

2. GAME OVERVIEW

Unexceptional.net is a mystical realist journey catalyzed by a series of interconnected events related to sexual infidelity, political conspiracy, and spiritual transformation. The project draws on the traditions of comics, graphic novels and computer games in order to create an environment that crosses boundaries between pop culture, fine art, and social critique. It also blurs the borders between “real” space and “virtual” space. The game has been developed as a net-centric, multimodal, pervasive action adventure RPG accessible via GPS enabled phones, the Internet, and a 3D game client.

In the game you get to play a supporting role to the main character, Guy who has discovered that his long-time partner is having an affair. This discovery launches him upon a series of quests that you participate in, in effort to gain insight into the nature of his partner's relationship. The main gateway to the game is through a Web portal designed by Guy, where he keeps a Blog documenting his daily trials and tribulations. He links to his comics, Web-hacks, and games from the Blog. Guy also provides running commentary on issues as his dramatic experience unfolds. Guy’s life is utterly out of control, and you attempt to help him regain a sense of stability.

For better or worse, Guy’s the kind of friend you like to have because he gives you and your other friends something to talk about. But unlike cults of personality built up around “live” celebrities where people must fantasize a personal connection to the star, Guy actually can reach people on a personal level. Moreover, he can do so on a non-human scale, because there's nothing to prevent him from carrying on thousands of intimate relationships at once, since for all practical purposes he's nothing but a highly scripted, automated and dynamically updated interactive database.

2.1 High Level Goals

There were a number of ideas motivating the development of the unexceptional.net project. These included an interest in linking a range of seemingly disparate devices and technologies together as part of a unified/shared game space; developing compelling content to demonstrate the power and potential of such linkage in a fun and playful way; and allowing players to not only become immersed in the environments created for them, but to participate in various ways to modify and/or extend that content based on their own interest and experience. Toward these interests, key objectives of the project included:

1) using unexceptional.net as a *testbed for deploying custom designed and freely distributed software* that takes advantage of everyday communication technologies such as Blogging, email, 3D gaming, and mobile telephony in order to enable anywhere anytime access to heterogeneous game worlds;
2) implementing the *game infrastructure* in such a way that it can either be *easily modified or used as a template* for alternative content development and deployment;
3) *facilitating ease of content creation* through provision of a Web-based "World-Building Toolkit";
4) *sharing the results in the public domain* through Internet distribution, formal exhibition in fine art contexts, professional conferences and events, and publication; and exploring novel forms of single-player and multi-player interaction.

2.2 Architectural Description

Figure 1 presents an architectural overview of the game and its components, which are rather extensive. All were either modified versions of existing open-source or freeware software, or our own custom solutions. They included:

- Greymatter[5], a Perl/CGI based flat-file open-source blogging software modified to dynamically load Guy's blog posts based on the player's game state;
- a 64 table MySQL[6]/PHP database for storing real-time game transactions including quest status and in-game item exchanges;
- a variety of web-based quests, and accompanying comic 'hacks' where comic panels were overlaid using DHTML on top of parsed and reconstructed web pages that players went to as part of quest completion requirements;

- a multi-modal location-aware phone game played in several different ways: (a) as a screen-based game using the number pad as a controller to navigate in-game guy's "physical body" from quest starting point to quest destination point; (b) by toggling to guy's "astral-body" mode which gets navigated in relation to the external game player's physical movement from a quest starting point to a quest destination point mapped in real-world space; (c) in "talk-mode" which interfaces an open source version of Asterisk's pbx telephony software integrated with CMU's Sphinx text-to-speech, speech-to-text software [7] to enable automated call routing to player's based on their physical location;
- a Java-based Applet simulator allowing players without GPS enabled phones to play the game via the project's web portal;
- a custom Java-based server to broadcast connection and location data to all active players so that information regarding which clients players were using to connect to the game was available, as well as their current game status;
- a 3D location-aware client using the Torque game engine [8] that essentially mirrored version (a) of the phone game making available 3D renderings of the game environment that also were keyed to player lat/lon location;
- a framework for shipping game world data to the cellphone and the 3D client from the central server based on the player's current region, enabling the phone game and Torque engine graphics to mirror the look and feel of real-world locations (i.e., if a player is over desert the game clients load desert terrain, if playing in a city, the clients show an urban environment, if played in coastal zone, coastal data is displayed, and so on);
- a pixel-based world map lookup table that allowed creation of "metaregions" (those regions defined as existing outside of the actual quest regions) so that the players have an approximate representation of the world no matter where they may be playing from;
- integration of the Googlemap API [9] to allow real-time tracking and display of all connected players, as well as summary game-state information, and ability to view prior quest paths stored in the database;
- PHP based modding utilities allowing customization of virtually all game assets in real-time through simple form-based submissions and/or data uploads;
- a Flash-based database transaction visualizer allowing dynamic viewing of other players' in-game status, including the ability to view players by quest status, geographic location, or needed items.

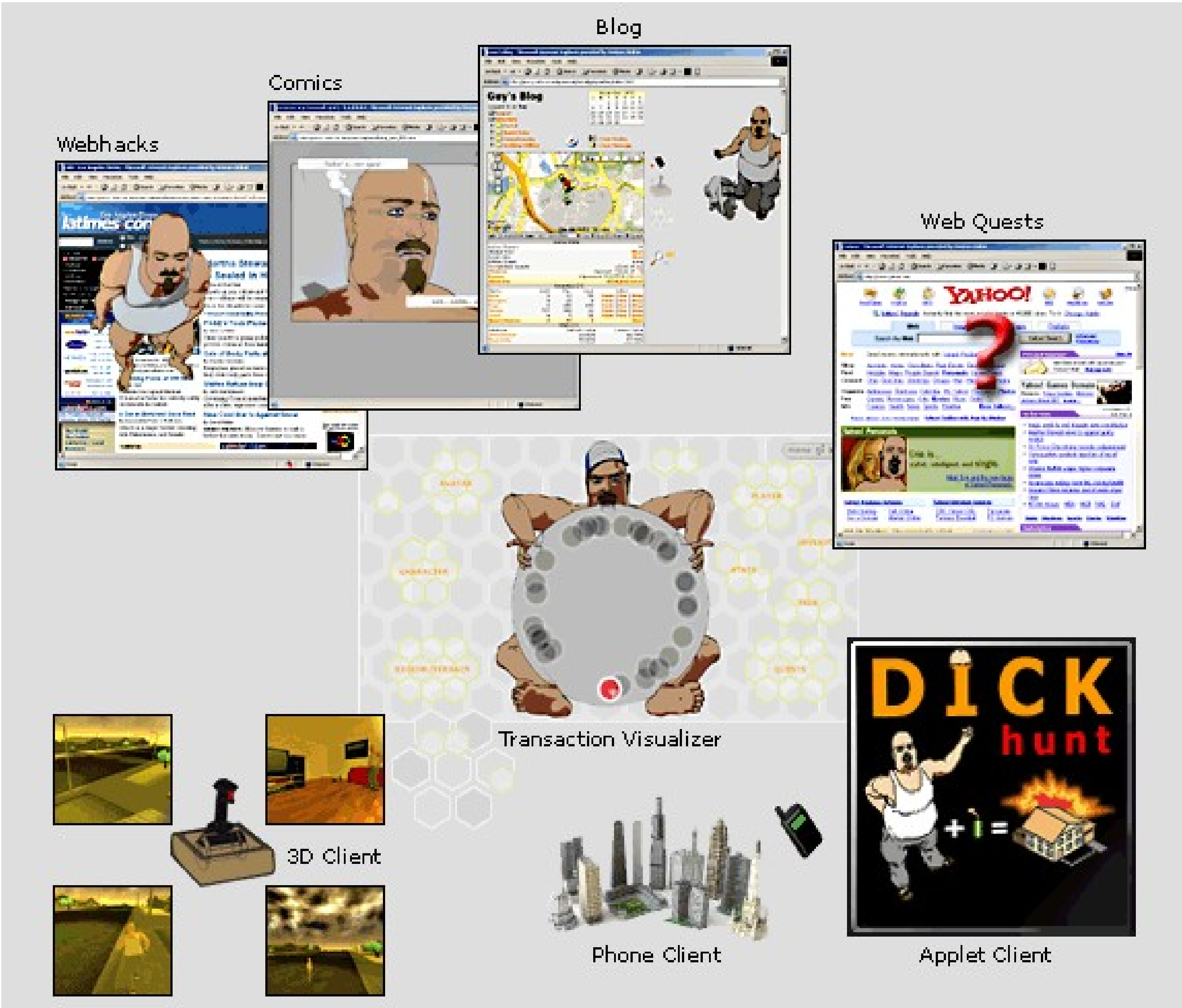


Figure 1 An Architectural Overview of unexceptional.net

2.3 The Narrative

To attain enlightenment Guy must be accompanied on a series of quests to find special objects that will help open all seven of his major "Chakras," the energetic centers of the body according to Buddhist doctrine.

The overall narrative arc entails:

1) an introduction to Guy as the crass, angry, resentful, cynical and curmudgeonly fellow that he is by nature;

2) the catapulting of Guy into a period of crisis linked to discovery of his cheating partner Betty and her love for Dick in the midst of terrorism, war, and homeland insecurity;

3) exposure to Betty and Dick's disturbingly co-opted Eastern mysticism, deployed as a means for experimenting with mind and body control techniques that use Guy as an unsuspecting guinea pig; and

4) Guy's neurotic compulsion to achieve enlightenment… whatever that may mean.

2.4 Example Scenarios

Greta, A PC user ends up at Guy's Web portal on her PC, reads a bit about his project, and decides to create an account in order to become a registered player. Account creation requires a player name, a valid email address, a mobile phone number, and a password. Upon registration she gets sent an email from Guy, and is forwarded to his Blog, which contains a single post providing context for the game about to unfold. The post also gives her the first quest, and provides a link for downloading Guy's recently released mobile phone game, "Dick Hunt" (see screenshot of the game in Figure 2). She activates the quest, and then downloads, installs, and launches the phone game.

When Greta starts the phone application, the entire game world – terrains, structures, characters, statistics, inventory, quest – gets built for her based on her geographic location. The game can now continue endlessly in every direction for Greta, due to an algorithmically generated grid-based game layout. Moreover, each grid has a simple coordinate that's stored in memory, which allows for identical path and object placement on return. The game also sends Greta's physical location information to the game server, allowing her to be tracked by other players in real-time. If Greta decides to play without a GPS enabled phone, or to simply use the Applet version of the game in Guy's online portal, she can still advance by:

1) *exchanging inventory items* with non-player characters in the Applet;
2) *offering to sell goods* to a shopkeeper accessible through the Blog; or
3) *participating in an online trading network*, also accessible through Guy's Blog, that allows her to post offers for goods to other players who are alerted via email as well as upon Blog login.

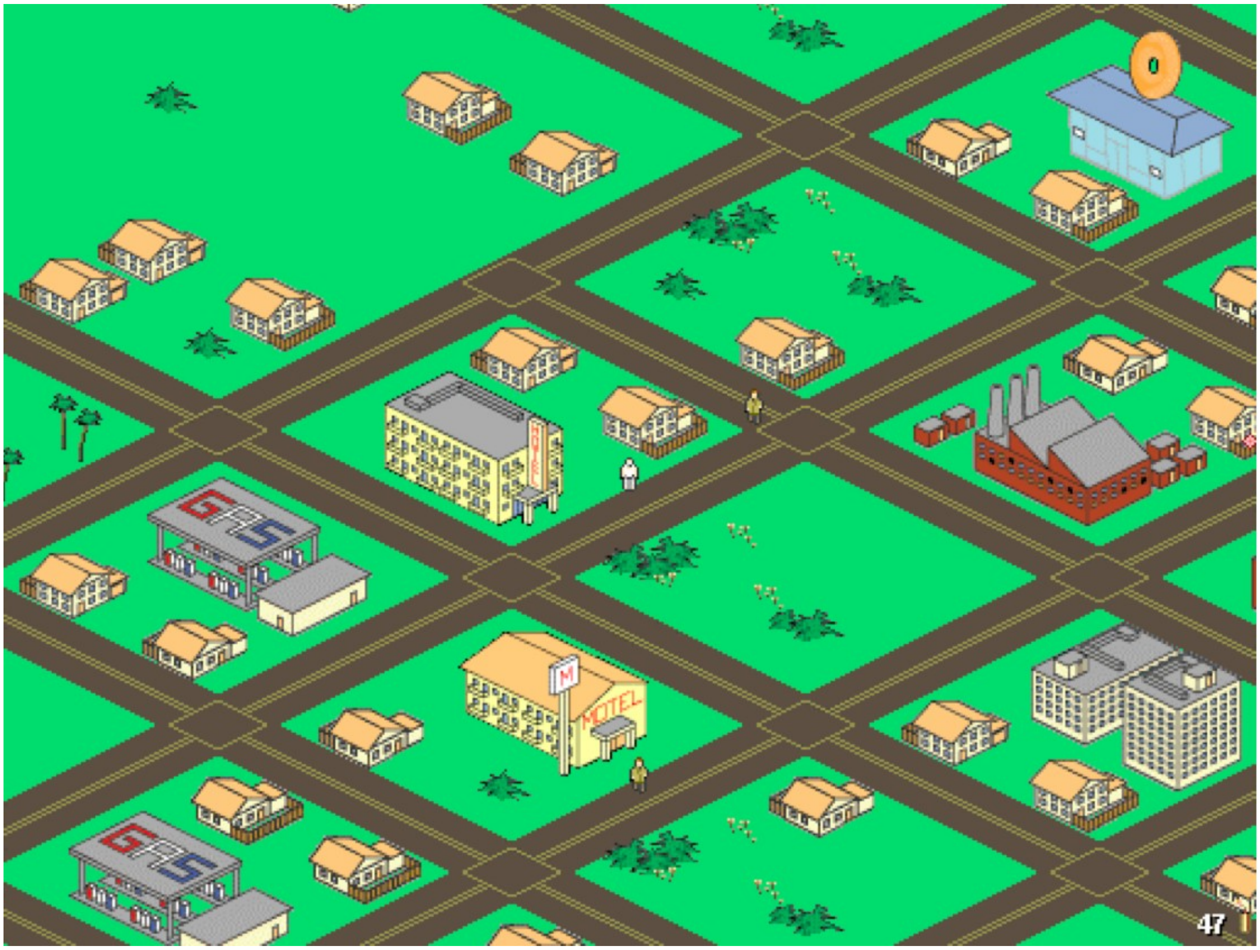


Figure 2 "Dick Hunt" phone game played in Guy's "physical body" mode

After several minutes Greta enters a pre-defined “hotspot” that causes an automated call to be placed from Guy’s help-bot to Greta. Greta’s phone rings, interrupting the visual interface to Dick Hunt. She answers, and can now continue her quest in voice-only mode. Guy’s bot tells her she’s in the vicinity of a spot where Betty was rumored to have spent time with Dick, and goes on to lists all the objects available for her to interact with, along with the actions that she can use to manipulate each object. Greta successfully “gets” some of the available objects, which get added to her inventory. She then unsuccessfully attempts to “use” one or two of them. Greta continues walking and talking, as her voice commands are interpreted on the fly by the text to speech and speech to text system. Along the way, she enters a region where another player is active. At this point Guy’s bot tells her that she may attempt to steal items from the inventory of the unsuspecting player. Greta does so, but unfortunately is unsuccessful, and instead has something stolen from her! The phone constantly updates the inventory and statistics kept in the database of both parties. Greta quits out of voice-mode and resumes playing the visual version of the Dick Hunt phone game. When she finally navigates both the in-game avatar as well as her physical body to the destination waypoint, she happily watches as a special key object descends from the heavens to be placed in her inventory.

Later Greta arrives back home and logs into Guy's Blog from her PC. She now sees her updated game-state information as well as a visual mapping of her movement in space and time. She also has a Blog-based link to a Web-page associated with the key object that contains a key-code that will allow her to gain access to critical game related information. Once accessed, her initial quest is completed, her stats are updated, and a new Blog post and quest are made available. Next time she thinks she may even want to try the 3D client (see screenshots in Figure 3). But for now, she's had enough of Guy and his chaotic world (see http://unexceptional.arts.uci.edu/).

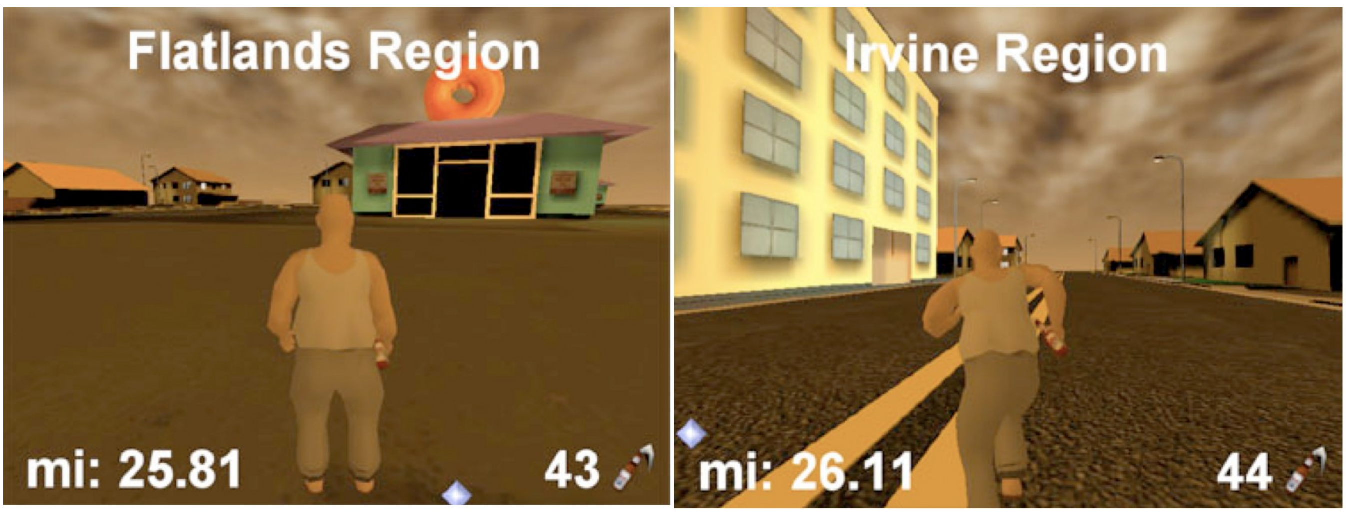


Figure 3 Torque Game Client 3D Region Samples

3. EXPERIENCES

In late Fall of 2002, Robert Nideffer started exploring ways to combine an interest in comics and graphic novels, computer games and the Web as a delivery platform for episodic content. The process began by playing around with a commercial 3D modeling software, Poser, to begin mocking up character designs. He developed a main character, "Guy", to serve as a proxy for exploring a range of issues related to interpersonal relationships, spirituality, and corporate and political interconnections related to developments in science funding and nanotechnology. From there he began to create and then narrate fairly conventional comic panels, that were laid out via DHTML, and clicked through as a series of layers that would be shown/hidden as users accessed them from a single Web-page.

At this early stage part of the interest was to investigate new forms of comic design and delivery. At the same time Robert had the idea of trying to use a newly released game development environment written by Ambrosia Software for the Apple platform called "Coldstone" [10] as another type of delivery platform. The idea was to create a top-down scrolling game, where the player moved Guy across the “pages” of the graphic novel/comic, revealing the comic narrative while playing an embedded game. Sadly, Coldstone development efforts stopped, and the engine was too unreliable to continue working with.

Shortly after, Robert visited the offices of Electronic Arts in Northern California to meet with Neal Young, one of EA's main game producers who had been responsible for directing a project called “Majestic” [11], an episodic, net-centric, conspiracy styled game that took advantage of a variety of everyday communication devices – Web, email, instant messaging, phones and FAX machines – in order to deliver game content. Majestic had the plug pulled on it for a variety of reasons after the World Trade Center bombings on 9/11. Robert's not so hidden agenda was to get EA to allow him access to the engine behind Majestic to the university for creative experimentation. Needless to say, that did not happen. Neal did indicate however, that the core engine technology behind the game, what they had termed the “experience server,” was not actually that complex – it was all the third-party deals that had to be struck to pull it off that made it so expensive (reportedly some $10 million). The outcome of that meeting was that Robert decided to independently pursue development of a similar framework for tying together heterogeneous devices as part of a shared game space, using free, part-time student labor at the university in the context of an art department, with no initial funding.

In 2003, Robert pitched the idea for “unexceptional.net” – a game environment that would be net-centric and have a range of modes through which the player could interact with it in real-time – to students in a course taught regularly in the school if information and computer sciences at UC Irvine called “Project in Software System Design.” That course required students to form teams of 3-4, bid on projects they found interesting, and create plans with the project “sponsor” (in this case Robert) for what they would achieve by then end of the term. The team's grade depended upon their success at achieving those goals. Three teams were interested in unexceptional.net. Normally only one team would be chosen per project, but given the complexity of the project and student interest, the instructor allowed two teams to begin working on it.

Over a two-year period, Robert worked with four teams of students from three different instances of that course. In addition, other students would hear about the project and become interested in getting involved. Thus, over a fairly extended period of time, different configurations of undergraduates, almost exclusively from the computer science department, worked on various pieces including early location-aware phone prototyping, database setup, PHP/MySQL integration, 3D Torque game engine scripting, 3D modeling for Torque, Asterisk PBX telephony and CMU's Sphinx integration for the talk version of the phone game.

This two-year phase was quite difficult, as teams would come and go. Some were talented and productive, others were not. There was little or no robust documentation or version control system in place. Thus Robert would have to walk new student collaborators through prior work, and figure out what to retain, and what to trash. In most cases, starting over from scratch became easier than retention; however it often took long periods of struggling to retain before coming to that realization.

Eventually, four of the undergraduate students (from the more than 20 that were involved at various points), began to play a key role as collaborators – Alex Szeto (a co-author of this paper, who came on board in 2004, and was responsible for the location-aware phone game, and the Java server, and whom Robert still collaborates with), Calvin Lee (who was responsible for much of the PHP/MySQL integration, Greymatter Blogware modification, and Asterisk/Sphinx integration), Dan Repasky (who helped with 3D modeling and scripting for the Torque game client), and Sunny Chu (who played a key role in scripting the Torque client version of the game). Two of those four (Alex and Calvin), Robert was able to hire on as paid part-time researchers in 2005, using money from a concurrent funded project. Once Alex and Calvin became formally involved, development moved forward on a far more consistent basis.

In addition to undergraduate collaborators, Robert worked with several graduate students – Sky Frostenson, Eric Cho, and Eric Kabisch – from the Arts Computation Engineering program that he co-directed. Sky and Eric had experience on prior projects that used the Torque game engine, and thus were in a good position to create the first prototype of the unexceptional.net Torque client. Sky and Eric were both key in mocking up the first characters, buildings, and environment (though neither were in a position to do any of the necessary game scripting). Prior to coming to ACE, Sky and Eric had also collaborated on a Flash based web game called “Bomb the Suburbs” that served as aesthetic inspiration for the phone game component of unexceptional.net. As a result, Sky also worked closely with Alex and Robert in creating art assets that helped to establish the look and feel for the mobile phone game.

Eric Kabisch had an interest in data visualization and a strong background in Flash programming, and in electronic music. A small amount of funding was secured to support Eric in developing a Flash component that would render all the database activity associated with unexceptional.net in real-time, with accompanying sound. Alf Inge Wang (co-author of this paper) has not contributed to the unexceptional.net project, but as visiting researcher at UCI initiated a process of publicizing results of this project in research papers.

As might be guessed, one of the main development difficulties had to do with managing a diverse group of largely unpaid students with varying levels of commitment over a lengthy implementation cycle lasting more than three years. Much of that time was spent “lost in code” meaning that difficult problems related to software architecture were being worked on across a range of technologies. During that phase, Robert worked on story content, user interface design across the range of devices, graphics for the phone, Web and 3D client, and Web design and implementation. Another main challenge was figuring out what was essential to the project, and what was extraneous. However the biggest issue had to do with the "creeping features" problem. Over the course of the nearly 4 years it was in development, unexceptional.net became so layered and complex that very few players actually had the chance to experience, understand or appreciate the scope and scale of the effort, and what was possible to do within it.

4. RELATED WORK

Although most pervasive games have a unique twist, there are several similarities of such games like the utilization of physical location and use of mobile technology. In this section we will describe some pervasive games described in the literature.

Cheok, Sreekumar, Lei and Thang describe a pervasive capture the flag game (CTF) [12]. The goal of the game is to use mobile devices, PCs and networks to provide an experience where the real-world is mixed with a virtual world. The setting of the game is medieval time where two teams fight as red and blue teams with the goal of capturing the flag at the enemy's castle (virtual castle). A team consists of two types of players: 1) real-world players representing knights that move around in the real world performing various actions; and 2) virtual-world players guides the knights through the real world based on a virtual map that describes various game elements such as castles, flags, bombs, traps, and magic potions. The flag is a physical wooden box holding a Linux-based Bluetooth device with a virtual representation, while the other entities are only virtual entities without any physical representation. The knights use a mobile phone with a GPS while the guides use PCs. All communication between knights and guides is performed as text messages within the game. Compared to the unexceptional.net game, this game draws clear boundaries between the virtual and the real-world through two different player types. In the unexceptional.net game these boundaries are blurred on purpose.

Epidemic Menace is a game where the players become medical experts with the goal of saving mankind that has been threatened by a mutated virus [13]. The setting of the game is that a mad scientist has created a lethal mutation virus, it has spread through out the university campus and infected all humans, and the players should through collaboration defeat this threat within three hours before it spreads outside the campus. Each team has a room equipped with stationary devices that allow players to observe and analyze the virus and to communicate with other team members. The mobile players have to go out to capture and destroy the virus. The mobile players are equipped with a PDA and a GPS device to track their position (position-belt) and either a mobile augmented reality system or a smart-phone. The mobile players can kill the virus by using a virtual spray they can use if they are close enough. The smart phone can be used to communicate with team game room or to catch the virus in the player's proximity. The game follows a pre-scripted plot. This game is similar to the CTF-game described in previous paragraph, but has also a story. However, the game is not story-driven but focuses on solving one specific problem.

SupaFly is characterized as a community-based virtual soap opera where the players create characters and interact through them [14]. The goal of SupaFly is to make and maintain relationships with other players through sending text-messages using SMS (short-message service). Further, a Web site is provided to let the players manage their characters and track their development in the game. The scores of the players are computed by analyzing how many relations a player have and how much this player has communicating to her/his friends. The goal of the game is to reach the highest level of status in the community to become “Supafly”. Every action in the game generates news in the game’s online magazine. The game also uses positioning to let the players pick up virtual objects by moving around in the physical world and if players are in the same area. SupaFly uses many of the same elements as unexceptional.net, but is less story-driven and mainly focus on social interaction.

moBIO Threat II Disease Control is a pervasive game similar to Epidemic Menace where two teams are competing to accomplish their own mission trying to prevent the other one from achieving its goals [15]. Terrorists have attacked and kidnapped the biologists at the university’s microbiology lab and stole their research. The goal of the game is to play either as terrorists or as a counter-terrorist that must prevent the enemy from achieving a mutilated pathological agent and lunch it using a missile. This can be done by either neutralizing the enemy force or by synthesizing an antidote for the agent being developed. The players use a Tablet PC with built-in loudspeakers, microphone, Bluetooth-card, and WiFi card. The WiFi network is used for communication between the players and the server, while the Bluetooth network is used to manage interaction between players that are physically in the same area. RFID tags are placed on physical objects being a part of the game such as trees and rooms. Location positioning is provided by mapping tagged objects in the physical world with their position.

The Drop [16] is another pervasive game where two teams use cellphones to play a variant of capture the flag where one team hides a virtual “briefcase” and the other team attempts to find it within a specified amount of time. The design of The Drop is based on the assumption that there is a beacon-based location system that can track mobile phones in the playing area. This means that the playing area must not be too large, e.g. a mall.

There are other examples of pervasive games made that we will not cover in detail such as a three-way physical game based on table tennis [17], PAC-LAN [18], Albert in Africa [19], Breakout for two [20], Swordplay [21], and Virtual aquarium and Mona Lisa bookshelf [22].

None of the game in this related work section provides the diversity and the multiple interfaces found in unexceptional.net. Unexceptional.net is also unique in the sense that it is strongly story-driven and that the story is dynamically revealed through a character’s Blog.

5. CONCLUSION

In this paper we have described the pervasive game unexceptional.net conceived and directed by Professor Robert Nideffer. The goal when developing this game was to explore ways to combine an interest in comics and graphic novels, computer games and the Web as a delivery platform for episodic content. The game communicate to the player through several channels like comic books, Blogs, Webhacks, Web quests, 3D-client, mobile client, Applet client, and a database transaction visualizer (see Figure 4). The game has built in location-awareness and all the parts of the game are put together through the story about a character name Guy. The game does not ask the player to solve one particular problem or task, but integrate the player in the story through various user interfaces. The game also utilizes interaction over the phone by using text-to-speech and speech-to-text software. The various software components in the game are based mainly on open source software integrated through a database. The unexceptional.net game shows how it is possible to produce pervasive game with a multitude of game-play elements and ways of interacting with the players.

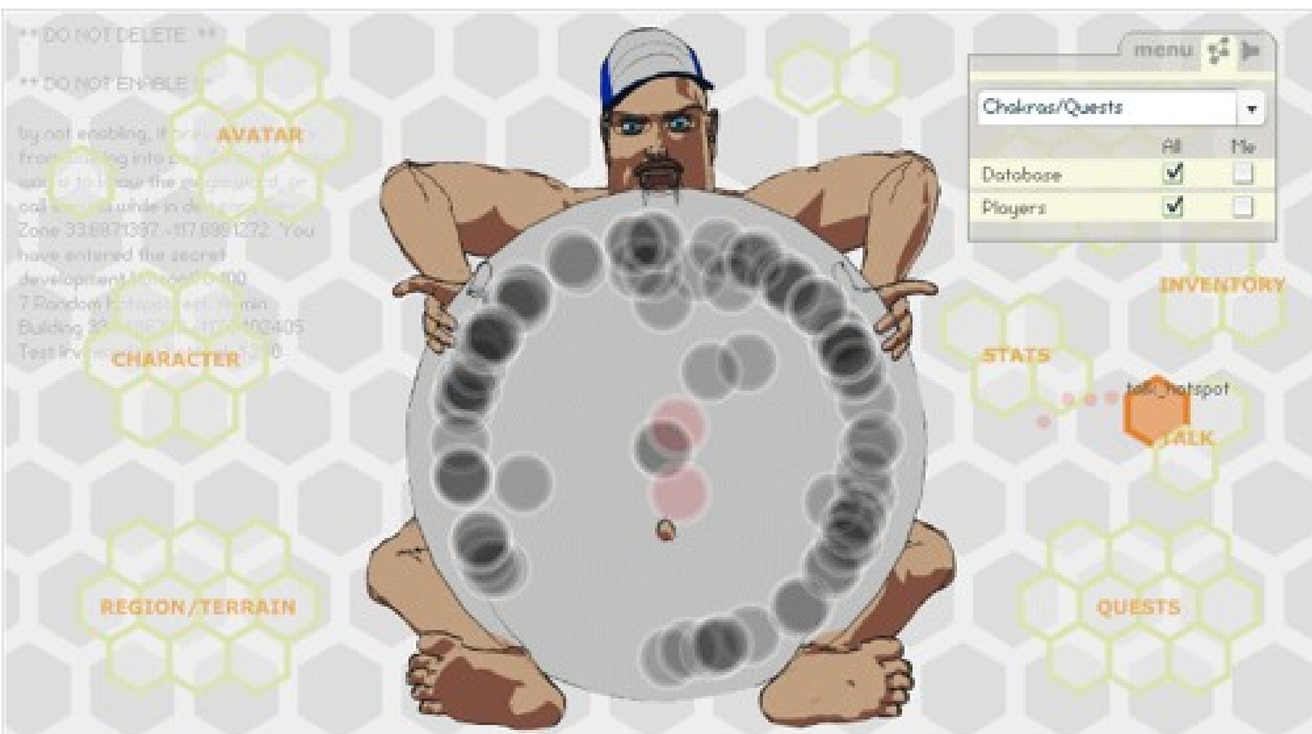


Figure 4 Database Transaction Visualizer

The experience from building the game described in this paper shows that it is hard to build such a game, as it consists of various loosely co-existing and interconnected parts that enable the diversity of gameplay and user interfaces. To develop such games in a university setting is especially difficult, as the project must rely on many short-time student projects where the developers leave after a relative short time. To succeed, it is necessary to have some dedicated developers over a longer period of time that understand the underlying architecture and know how to get the different parts to communicate. One of the main challenges of building a pervasive gaming with multiple interfaces and gameplay elements is to establish an architecture that allows the game to change in any direction (evolve), and at the same time provide robustness and a system that is easy to configure. Diverse pervasive games have to cope with configuration challenges in making various technologies work together. This is always a difficult problem, as new releases of underlying third-party components will cause compability problems for the whole game.

The unexceptional.net game shows that pervasive games can give new and unique game experiences that involve other user interfaces than the ones normally used for gaming. The future will reveal new pervasive games that will take players to new places never seen before and let them experience a game through multiple interfaces and gameplay elements.

ACKNOWLEDGMENTS

We would like to thank all persons that have contributed to the unexceptional.net project, and special thanks go to Calvin Lee, Dan Repasky, Sunny Chu, Sky Frostenson, Eric Cho and Eric Kabisch.

REFERENCES

[1] Amory, A., Naicker, K., Vincent, J., and Adams, C. 1999. The use of computer games as an educational tool: Identification of appropriate game types and elements. British Journal of Educational Technology, 30(4): 311-321.

[2] Malone, T. W. 1981. Toward a theory of intrinsically motivating instruction. Cognitive Science, 4(13): 333-369.

[3] Chen, V. H. and Duh, H. B. 2007. Understanding social interaction in world of warcraft. In Proceedings of the international Conference on Advances in Computer Entertainment Technology, June 13 – 15: 21-24.

[4] Magerkurth, C., Cheok, A. D., Mandryk, R. L., and Nilsen, T. 2005. Pervasive games: bringing computer entertainment back to the real world. ACM Computer Entertainment, 3(3), July.

[5] Noah Grey. 2008 Greymatter – The Opensource Weblog/Journal Software. Web: http://noahgrey.com/greysoft/, Accessed Dec 19th 2008.

[6] Digium. 2008 Asteriks – The Open Source PBX & Telephony Platform. Web: http://www.asterisk.org/, Accessed Dec 19th 2008.

[7] SourceForge.net. 2008 CMU Sphinx. Web: http://sourceforge.net/projects/cmusphinx/, Accessed Dec 19th 2008.

[8] GarageGames. 2008 Torque Game Engine. Web: http://www.garagegames.com/products/torque/tge/, Accessed Dec 19th 2008.

[9] Google. 2008 Google Maps API – Google Code. Web: http://code.google.com/apis/maps/, Accessed Dec. 18th 2008.

[10] Wikipedia.org. 2008. Coldstone game engine. Web: http://en.wikipedia.org/wiki/Coldstone_game_engine, Accessed Dec. 18th 2008.

[11] Wikipedia.org. 2008 Majestic (video game). Web: http://en.wikipedia.org/wiki/Majestic_(video_game), Accessed Dec 18th 2008.

[12] Cheok, A. D., Sreekumar, A., Lei, C., and Thang, L. N. 2006. Capture the Flag: Mixed-Reality Social Gaming with Smart Phones. IEEE Pervasive Computing, 5(2): 62-69.

[13] Lindt, I., Ohlenburg, J., Pankoke-Babatz, U., and Ghellal, S. 2007. A report on the crossmedia game epidemic menace. ACM Computer Entertainment, 5(1), Jan.

[14] Jegers, K., and Wiberg, M. 2006. Pervasive Gaming in the Everyday World. IEEE Pervasive Computing, 5(1): 78-85.

[15] Segatto, W., Herzer, E., Mazzotti, C. L., Bittencourt, J. R., and Barbosa, J. 2008. Mobio threat: A mobile game based on the integration of wireless technologies. ACM Computer Entertainment, 6(3), Oct.

[16] Smith, I., Consolvo, S., and LaMarca, A. 2005. The Drop: pragmatic problems in the design of a compelling, pervasive game. ACM Computer Entertainment, 3(3), Jul.

[17] Mueller, F., Gibbs, M., Vetere, F., and Agamanolis, S. 2008. Design space of networked exertion games demonstrated by a three-way physical game based on Table Tennis. ACM Computer Entertainment, 6(3): 1-31.

[18] Rashid, O., Bamford, W., Coulton, P., Edwards, R., and Scheible, J. 2006. PAC-LAN: mixed-reality gaming with RFID-enabled mobile phones. ACM Computer Entertainment 4(4), Oct.

[19] Newman, K. 2005. Albert in Africa: online role-playing and lessons from improvisational theatre. ACM Computer Entertainment 3(3), Jul.

[20] Mueller, F., Stevens, G., Thorogood, A., O'Brien, S., and Wulf, V. 2007. Sports over a Distance. Personal Ubiquitous Computing 11(8):633-645.

[21] Katzourin, M., Ignatoff, D., Quirk, L., LaViola Jr., J. J., and Jenkins, O. C. 2006. Swordplay: Innovating Game Development through VR. IEEE Computer Graphics Application 26(6): 15-19.

[22] Nakajima, T., Lehdonvirta, V., Tokunaga, E., Ayabe, M., Kimura, H., and Okuda, Y. 2007. Lifestyle Ubiquitous Gaming: Making Daily Lives More Pleasurable. In Proceedings of the 13th IEEE international Conference on Embedded and Real-Time Computing Systems and Applications, August 21 - 24

Spew

(A Work In Progress)

'08-'09

Core Zone At Night - Gas and Oil Data

- **Comprised of three economic zones that comprise the "World System" - the Core, the Semi-Periphery, and the Periphery**
- **Utilizes a variety of static and realtime data streams to drive the simulation game environment**
- **Streams zone specific time of day, traffic, population, health, economic, ecological, news media, twitter feeds, and various other data**
- **Built in tutorial system**
- **Incorporates animated objects**
- **Navigate via Googlemap style click and drag, as well as by vehicle travel**

Core Zone At Night - Bank Data

- Clicking objects in zones accesses data linked to them
- Player must make decisions in relation to data and events spawned in-game
- Player (mis)manages capital over time in effort to balance civic, economic, and ecological health
- Decisions made in one zone have impact across all zones

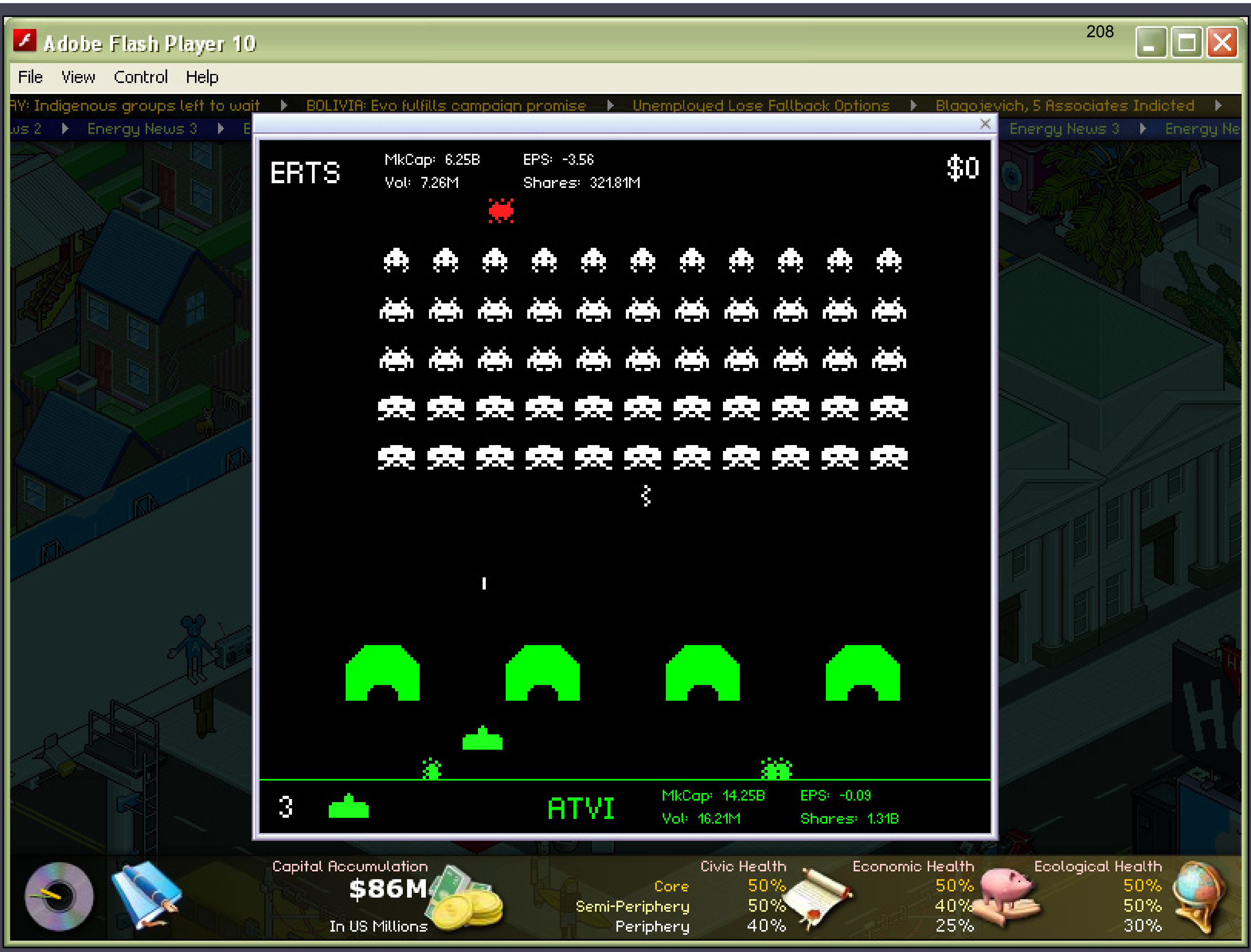
Core Zone - Stock Invaders Minigame: Stock Selection

- **Zones have minigames that also impact capital management**
- **Stock Invaders is a Core Zone minigame**

Core Zone - Stock Invaders Minigame: Stock Summary

- Different stocks are chosen to represent the player and the invaders
- Game difficulty and object attributes scale in relation to real-time market data

Core Zone - Stock Invaders Minigame: Gameplay



- **Faithfully modeled after original Space Invaders videoarcade game**

Periphery Zone - Breaking News Event

- In-game events requiring action spawn in relation to player's decision making
- Game engine supports both in-game generated news and world news streamed in from external feeds

Periphery Zone - Medical Data

- **Encourages comparative data analysis as part of play**

Semi-Periphery Zone - Medical Data

- **Feeds locationally / topically / linguistically specific data into zones**

PORTAL

Decentralized Virtual Activities and Technologies: A Socio-Technical Approach

Walt Scacchi, Alfred Kobsa, Crista Lopes, Gloria Mark, Bonnie Nardi, David Redmiles, and Richard N. Taylor

Institute for Software Research
University of California, Irvine
Irvine, CA 92797-3455 USA

December 2008

Our proposal is focused on the topic of decentralized virtual activity systems (DVAS) employing an empirical socio-technical research approach. We see growing, widespread interest in the development and use of decentralized systems and virtual world environments as possible new places for engaging in collaborative work activities. The Gartner Group recently declared that within five years 80% of Internet users and Fortune 500 companies will have an online presence in a virtual world of some kind. Elsewhere, there is widespread interest in stimulating new technological innovations that enable people to come together through social networking, file/media sharing, and massively multi-player online game play. This new generation of networked computing environments seems headed towards increased socialization, interaction, communication, and collaboration that span multiple organizational boundaries as its primary purpose. But how do we get there from here? Is it sufficient to just let the market of entrepreneurial vendors and technological innovators simply decide who needs what? The history of computing reveals a legacy of many failed or problematic efforts to develop and deploy computing systems that arise from a lack of understanding or recognition of the ways how people's work and social activities are situated in organizational and technological contexts. These contexts configure, constrain, or enable some types of activities to flourish while others are displaced, either unintentionally or intentionally.

DVAS requires a broad interdisciplinary understanding of the problems and a broad and interdisciplinary approach to their solution. We are an interdisciplinary team comprised of researchers capable of investigating socio-technical issues of collaborative systems. We are proposing a large multi-site, multi-partner research endeavor that provides the greatest practical opportunity for generalizable results from comparative analyses of both in-situ field studies and technology prototyping efforts. Our five research partners serve multiple roles in this project. First, they *serve as a source of real-world problems* for us to tackle. Second, as organizations facing the daily problems of distributed development, our partners view these as practical problems to be tackled, and *will engage directly with us in developing strategies and solutions*. Third, they *serve as test-beds* for early evaluation of proposed new solutions. In other words, by working with our research partners, Aerospace Corporation, Avaya Labs, Discovery Science Center, Northrop Grumman, and Unimodal Inc., we ensure a continuous engagement with real world settings at all stages of the project. Thus we have access to real-world workplaces and people who are confronting the kinds of problems and issues we seek to empirically investigate.

Intellectual Merit: The core of our research is focused on both the individual and collective study of six critical variables: representations and realities, conflicting policies and practices, relationship work, processes and coordination, privacy and awareness, and security and trust. No prior study has undertaken such a comprehensive range of socio-technical variables as a multi-faceted lens through which to examine emerging computing system technologies in a multitude of settings with diverse stakeholders. However, based on the results of our prior NSF study on the development and use of distributed, inter-organizational systems, we believe our team is in the right place conceptually and theoretically to conduct the study, perform the necessary comparative analyses, and produce results tgeneralizable, with appropriate qualifications, to other settings.

Broader Impacts: We believe that research on DVASs will have a vital impact on society. As development and use of DVASs becomes more common practice and as organizations continue to become more decentralized, new methods and policies will need to be identified and tested to enable people to collaborate successfully. We also believe that our study has economic value as it will help organizations to carry out decentralized work effectively with smoother coordination, so that they can better compete in the global market. DVASs will be resilient to environmental disruptions as collaboration will be able to be conducted from anywhere, anytime, using representations of people, artifacts, and activities Understanding the unique challenges of work in decentralized and virtual activities, as well as designing DVAS technologies to meet the challenges, will have both theoretical and practical impact. Theoretically we want to determine the contours of decentralized activity, allowing comparison to other social forms such as traditional hierarchies, communities of practice, and rational bureaucracies. Finally, our results will also impact higher education as new people entering the workforce will need to gain skills in developing DVAS, and in working in decentralized virtual worlds and environments.

1. Introduction and motivation

We see growing, widespread interest in the development and use of decentralized systems and virtual world environments as possible new places for engaging in collaborative work activities. The Gartner Group [1] declares that within five years 80% of Internet users and Fortune 500 companies will have an online presence in a virtual world of some kind. Furthermore, they estimate the collaborative and community-related aspects of these environments will dominate in the future. Elsewhere, there is widespread interest in stimulating new technological innovations that enable people to come together through social networking, file/media sharing, and massively multi-player online game play. This new generation of networked computing environments seems headed towards increased socialization, interaction, communication, and collaboration that span multiple organizational boundaries as its primary purpose. But how do we get there from here? Is it sufficient to just let the market of entrepreneurial vendors and technological innovators simply decide who needs what? The history of computing in organizational and work settings reveals that the adoption and integration of new technologies is rarely simply a matter of buying the lowest cost technical alternative. More often such history reveals a legacy of many failed or problematic efforts to develop and deploy computing systems that arise from a lack of understanding or recognition of the ways how people's work and social activities are situated in organizational and technological contexts. These contexts configure, constrain, or enable some types of activities to flourish while others are displaced, either unintentionally or intentionally. Our goal in this proposed study is to empirical investigate and comparatively analyze the development and use of decentralized virtual activities systems (DVAS) within a sample of five diverse organizational settings using a research approach that foregrounds a balanced socio-technical perspective.

Our five research partners serve multiple roles in this project. First, they *serve as a source of real-world problems* for us to tackle. Rather than speculating about the problems of the development and use of DVAS, or working with abstract theoretical models, we will study how DVAS arise in practice in real settings. Since the social and organizational context in which development takes place is a critical aspect of a successful development model, this sort of engagement is essential. Second, as organizations facing the daily problems of distributed development, our partners view these as practical problems to be tackled, and *will engage directly with us in developing strategies and solutions*. Third, they *serve as test-beds* for early evaluation of proposed new solutions. As organizations with real and pressing problems surrounding inter-organizational work, our partners provide ideal settings in which to evaluate research outputs *in situ*, providing more rigorous evaluation than would be possible in laboratory settings or simulations. In other words, by working with our research partners we ensure a continuous engagement with real world settings at all stages of the project.

Beyond this, we believe a large multi-investigator effort is required to conduct such a study. Why? First, a study of DVAS requires a broad interdisciplinary understanding of the problems and a broad and interdisciplinary approach to their solution. We are a unique interdisciplinary team comprised of researchers with years of scholarly and professional expertise that we have applied to investigating socio-technical issues of collaborative systems. We are ideally positioned to study the issues critical to DVAS. We are proposing a large multi-site, multi-partner research endeavor that provides the greatest practical opportunity for generalizable results from comparative analyses of both in-situ field studies and technology prototyping efforts. A small/mid-size team does not have the resources or expertise to realistically study and develop prototypes for DVAS in such varied organizational settings. Only a large team has the potential to realize all of the preceding opportunities, benefits, and constraints; we assert that no amalgamation of individual investigator studies can cover the same amount or diversity of research interests or research partners or provide such comprehensive expertise in ways that can more effectively achieve comparable results. Accordingly, we now turn to describe and explain our proposed research effort, starting first with some background that helps further identify the focus of our study.

2. Background

What is a decentralized virtual activity system? First, an *activity system* is a computer-based environment that encompasses a web of participants, processes, information artifacts, information technologies, products and services, customers/users, organizational setting(s), as well as human, technological, and resource infrastructures [50] [107] that are interrelated in ways that can enable workplace or other activities [48]. An activity system is a contemporary *socio-technical system* whose social and technical elements are interdependent and co-evolving, so that attempting to discount/ignore either the social/technical dimensions results in ineffective systems that are poorly understood and wasteful of scarce resources. An activity system is similar in concept to a work system [6], but we do not see it as limited to only routine business processes or other work activities in an enterprise setting, since other productive activities including building social relationships, playing games, or engaging in exploratory experiences are all within our view of an activity system.

Second, a *virtual activity system* (VAS) is one where some of the elements of an activity system either exist or are accessible only in a computer-mediated form, and may be distributed across multiple sites/computers. Text-based virtual realities (also called Multi-User Dungeons or MUDs [20], virtual worlds [98] [8], networked virtual environments [87], and persistent online games like *World of Warcraft* or *City of Heroes* are all kinds of VAS, as are computer-based models or simulations of enterprises as rendered within an Enterprise Requirements Planning (ERP) system, computer-aided manufacturing systems, multi-agent problem solving systems, and virtual organizations. The vast majority of open source software development projects primarily rely on text-based communications through "informalisms" like threaded email discussion lists, project wikis and instant messages that are supported by ad hoc arrays of personalized software or artifact development tools, to enable decentralized software systems and development processes [82] [83] [85]. In a sense, these OSS developers work in virtual worlds of online text, communication and discourse through informalisms that move in, out, and across both shared and private information repositories, as well as across organizational boundaries. The multi-party telephone conference call (with or without Web-based supplements and presentation materials) are common, widespread, and often ready-to-use technical systems that enable work to be decentralized and globally dispersed, are also considered (early) virtual worlds. On the other hand, the Web is generally not identified as a virtual world, nor is the use of email, nor is an integrated development environment with shared repositories for building software. Yet each of these can be very effective in supporting technical work activities, software system design, or system development processes that can be logically centralized and physically distributed. As such, a VAS is one where work activities occur via creation and update to information artifacts that move in and out of information repositories and the communications among participating contributors, so that these work activities can be physically distributed but logically centralized and coordinated [64]. Thus, VAS allow for development and use of virtual worlds based on text, graphic, electronic media, databases, or some combination thereof. Similarly, VAS allow for workplaces that exist only online and lack a physical workspace or specific setting.

Next, a *decentralized activity system* (DAS) is one where elements of the computer-based environment are not only physically distributed across multi sites, but also where administrative or social control, allocation of resources, and decision-making activities with regard to how the DAS is developed or used are performed locally in an autonomous or discretionary manner (at least to some degree). DAS both seek to engender and accommodate heterogeneous computing systems that can interoperate at arms length via information sharing protocols, processes, and data representations. Both the Internet and World-Wide Web were implemented as DAS, as are other computer-based environments for peer-to-peer applications like file sharing or resource-sharing. Virtual organizations, through communication, data-sharing and coordination technology support, have been able to respond to dynamic changes in the environment by adopting decentralized structures [5] [9] [23] [22] [49]. DAS must self-organize, and DAS developers and users must organized themselves into participatory roles, role-sets and role-migration paths so that the DAS can persist and thrive as participants join and leave the effort. Collective activities that span across the weak social network ties can nonetheless give rise to a global social movement [11] or computerization movement, like the Free Software Movement [30, 32].

Finally, a *decentralized virtual activity system* (DVAS) is an activity system whose elements can be both virtual and decentralized. Here we look for emerging computer-based environments that seek to create or articulate a virtual organization or virtual world that spans multiple interacting organizations with diverse stakeholders (who may be in conflict with one another, yet need to cooperate or coordinate their activities) that seek to collaboratively interoperate as a loosely coupled alliance, rather than as a hierarchically controlled network of administrative authorities or resource controllers. In conjunction with our research partners, described later, we have identified five different settings where DVAS are being developed and used, or where efforts are underway to create and employ a DVAS.

Decentralization is not inevitable, but it has the potential to be very effective in many situations, as do centralized systems and administrative regimes with distributed developers and end-users. Similarly, highly visual virtual worlds may or may not be the workplace of the future for everyone. Subsequently, we seek to understand where, when, how, what kind, and why decentralization works best, and whether or not decentralized virtual worlds are a necessary or useful part.

Overall, it remains unclear as to what principals, models, or guidelines people can employ to help determine whether/how their technical work activities, software system designs, or development processes might best take advantage of the kinds of decentralization that we have identified. Similarly, people need guidance to recognize when such efforts like the move to virtual worlds of one kind or another are a good/bad fit, unnecessarily complicated, or otherwise inappropriate choice that might otherwise be better served through alternative means. Our proposed research investigation seeks to discover and refine such principals, models and guidelines based on both empirical field studies and experimental system prototyping efforts to ground are findings. We similarly seek to demonstrate and iteratively assess how such results can be adopted, assimilated, and refined into work practices in the enterprises of our research partners.

Research objective: Our research objective is to understand the constraints, properties, opportunities, and affordances associated with different DVASs, as well as understanding what practices, guidelines, principles, models, and theories can best shape the development and use of a DVAS in different types of settings. For example, a new startup venture like Unimodal Inc. chooses to have a research team at UCI prototype a new activity system for simulating a next-generation personal rapid transit (PRT) system using a virtual world platform like *Second Life*. Furthermore, they both want to allow or encourage other users in *Second Life* to try out the new simulated PRT system to see what users learn from their interaction with each other when exploring the new system. In so doing, Unimodal and the UCI researchers subsequently seek to link activities within the simulated virtual world to digital objects (PRT passenger pods, PRT network rail switches, embedded sensors, etc.) or events distributed in the physical world of an actual PRT system prototype (soon to be built), as well as how these users and the system's developers make sense of their individual and shared experiences with the physical/virtual elements.

Figure 1: Simulated PRT system modeled in *Second Life* (from UCI Prof. C. Lopes and colleagues) on left, PRT system prototype envisioned in physical location (from Unimodal Inc.) on right.

In our proposed study, we will engage a series of field studies and prototyping experiments with five research partners including Unimodal Inc. that will provide empirical observations, support comparative analyses, and enable formulation of principles, methods and theoretical models that generalize across our study samples so as to inform future developers and users about how best to build and employ new DVASs in socio-technically effective ways.

We now turn to describe some of the related research that informs our proposed research study and variables.

3. Research variables

Our proposed research effort is focused understanding how systems supporting decentralized virtual activities and technologies (DVAS) are developed, used, and adapted in diverse settings. From this, we see an array of variables that span a socio-technical landscape of alternative choices for action and system design. These variables (or sets of related variables) range from (a) how to bring together diverse stakeholders in immersive worlds in ways that enable mutual sense-making and dialectical evaluation of new DVAS applications through alternative *representations that span physical and virtual realities*; (b) *how people who collaborate through DVAS create new policies and practices that engender or mitigate conflicts* in the resulting work practices, which in turn shape and guide their primary work activities; (c) *how people engage the work of creating, sustaining, and adapting relationships in DVAS environments* that most effectively meet their needs; (d) how can geographically dispersed people articulate development processes, work practices, and project communities for DVAS; (e) how people can *manage or control their privacy concerns as well as how such concerns are exposed to others* when interacting through artifacts or avatars; and (f) how to define and provide *technical foundations for security and trust in DVAS*.

Each of these issues can be briefly described in turn as follows.

Representations and realities: A primary attraction of DVAS is the ability to blend real artifacts and people with virtual ones to accomplish, for example, a design task. An instance of this is seen in our background work in the design of a rapid transit system: engineering specifications of the real artifacts are used in a virtual environment to explore design choices. Of course the real artifacts are not actually used; representations of them are. The goal of this aspect of our research is to understand the issues and trade-offs involved in designing representations of non-virtual entities for use in a DVAS, yielding principles for guiding developers in creating new ones.

Research questions that arise from this goal include the following. First, what are appropriate ways to represent people, especially people playing specific roles? For instance, in a decentralized software development project, an individual may play a key role as the configuration manager. Should the representation focus solely on this role, or should characteristics perhaps not directly related to the official role be included? How does the social web of developers in a project affect appropriate choices for representation? Second, when should representations be used? With what fidelity? What kind of flexible boundary can be used between the two? For instance, in the software development project, when is it more effective to manipulate representations of the software artifacts as opposed to the artifacts themselves? Or should all meetings have to take place in a virtual setting? For all participants? Third, when are "representations" appropriate which do not have any natural counterpart in reality? For instance, composite or intangible entities, such as "this software project" could have an explicit representation in a DVAS.

In short, we want to find out to what extent the different stakeholders of a complex infrastructure change can come together in an immersive DVAS world like those enabled in *Second Life*, and whether such common "live" representations can support mutual understanding of the system and its broader consequences.

Conflicting policies and practices: Our goal is to study how a socio-technical approach can enable people who are collaborating in a highly decentralized work structure to create new policies and practices that relate to their new work practices, in order to achieve benefits for the decentralized team and organization.

The challenge for reconciling different practices is that there is no standard approach in a decentralized setting, as there would be in a traditional hierarchical, centralized structure. People must find a way to articulate differences and self-organize to set new standards and conventions of practice. To achieve this, different perspectives must first be made explicit [88] [86] [54]. Focusing on the shared objects used in decentralized work practices is a way to understand the relationship of work routines and practices, and work products. Bergman et al., [12] discovered that these boundary objects can be successfully used as active facilitators to help distributed workers develop shared and congruent perspectives and policies by serving to promote shared representations, transform knowledge, mobilize for action, and legitimize new knowledge. We will investigate how technology can enable perspectives to be linked to such boundary objects in decentralized work forms to assist people in transforming their practices to achieve congruency across distance.

Relationship work: We seek to understand how people create and sustain relationships in decentralized activities in order to design technologies to more effectively support relationship work. We will examine the shape and dynamics of social networks, the means by which people contact and exchange information with one another, formal and informal mechanisms by which relationships are created and sustained, and what is unique about relationships in decentralized activities. The goal of this research is to provide broad understandings that will enable people in DVAS to examine and interpret their own practices surrounding relationship work as well as to inform the design of new technologies that will be undertaken as part of the proposed research. This is consistent with a socio-technical approach in which both practices and technologies are seen to mutually and dynamically affect and transform one another. Recent research suggests that social networks in distributed organizations are held together by small cliques that work across the boundaries of the groups within the networks [109]. We will examine network dynamics to understand how specific relationships, such as those in cliques, influence the function of social networks in DVAS. We are also interested in the whole suite of technologies people use in relationship work [61] and the ways in which new technologies, such as those we develop, will alter such practices. It is also important to remember the possible role of face-to-face communication in DVAS. Although they are primarily distributed, some critical face to face interaction may take place and, in line with our socio-technical approach, we will be alert to understanding its possible role in activities that are primarily conducted in virtual space [62].

Discovering DVAS development processes, work practices, and project community dynamics: We seek to develop empirically grounded models that account for the conditions, circumstances, and events people working in formal organizations or in ad hoc communities when they address whether to: (a) embrace free/open source software (FOSS) or proprietary development techniques as a major mode of system development of DVASs? (b) enable the creation and deployment of decentralized virtual environments that are built from FOSS versus proprietary components, systems, or applications? (c) enable end-user developed DVAS application systems or systems for sharing user-created open content? and (d) enable socio-technical networking of DVAS across enterprise boundaries to form alliances with external partners or competitors? The resulting models will help identify fundamental principles, properties, and theories accounting for DVAS development and use.

Managing privacy and maintaining awareness of impressions: Our research on impression management for improving privacy management will address the following research questions: How can the impression(s) one conveys through DVAS systems be made more visible to oneself? What DVAS mechanisms can empower users to manage their impression(s) appropriately? How is appropriateness of impression(s) interpreted and evaluated? How can DVAS mechanisms be made to integrate seamlessly with user practices to avoid undue burden on users? To what extent do DVAS systems with improved impression visibility and management succeed in improving privacy management?

Security and trust: The absence of a central, trusted authority in DVASs (a consequence of decentralization) implies that they are potentially subject to attack or inappropriate use. Our goal is to understand how to make DVASs useable in a world where malicious behavior is to be expected. In absence of such understanding, the best DVAS technology will never be used in organization-critical applications. Three research questions follow from this goal. (1) What are the risks most critical to DVASs? Scoped more narrowly, what are the risks critical to the DVAS of our research partners? (2) Can previously identified mechanisms for mitigating trust and security risks satisfy the identified needs of DVASs? If not, how not, and why? (3) What new security and trust mechanisms can augment the prior work and be effectively incorporated into DVASs, to satisfy any newly identified needs?

4. Insights from Prior Research

Each of our six research variable sets builds from a legacy of prior research that our project investigators have performed, as well as related research from others. These insights help to further explain why we have selected each of the research variable sets introduced in the previous section. Accordingly, we now review what we already have learned about each of these issues for further investigation in this proposed effort.

Representations and realities: Activities around deploying new technologies always involve a diverse collection of stakeholders to come together in a fairly decentralized manner. One good example is Personal Rapid Transit (PRT). Because of concerns with the environment PRT has recently got considerable attention from the part of local and central governments all over the world. PRT is has the potential to drastically reduce air pollution and redefine public transportation. Although the interest in PRT has been growing, its deployment in real settings involves a convergence between technologists, politicians, service providers, and, especially, it has to have a broad support from the general public. All these stakeholders have their own set of concerns and political power, they are by nature decentralized, and their coordination for a common goal is a major hurdle.

Since PRT is a radical departure from the current transportation infrastructure, including self-driving vehicles that carry people, there is an uphill battle for its adoption. The different stakeholders often express concerns such as safety (fail-safe control of the vehicles in the guideway network), visual pollution from having to build guideways, and whether it's practical at all to embed such systems within cities, and use them on a daily basis.

One of the PIs has been working with a company in Southern California, Unimodal Inc. that is trying to deploy one such PRT system. As part of this collaboration, the PI developed a technically-accurate simulation of their PRT system using the virtual world *Second Life* [59] [53]. The guideway and control of the cars was developed according to the actual specifications given to us by the company's engineers.

We found numerous advantages in doing this simulation in an immersive 3D environment like *Second Life*. Not only were technical design problems uncovered, but several issues pertaining to usability and adoption emerged [53]. The fact that the simulated system is publicly available makes it easy for anyone – politicians, the media, and the general public – to experience it to some extent before it is deployed in the real world. This immersive experience can be done in a decentralized and relatively chaotic manner, at the stakeholders' own paces. But the virtual deployment site is also a common point, a common representation, which the stakeholders can refer to, and experience.

The insight we gained is that publicly accessible virtual worlds can be powerful tools not just for engineering design, but also for supporting the natural decentralization of stakeholders in complex techno-social situations.

Conflicting policies and practices: The interoperability of systems to support collaboration requires moving beyond purely technical issues; it also concerns the means and practices that users adopt to carry out their cooperative activities [86]. Decentralization compounds the interoperability problem as it introduces a new kind of interaction order. People who work in decentralized configurations have developed unique practices in their local work settings, and must follow locally established governmental or organizational policies. At the same time they interact in decentralized work structures, networked electronically across distance. This duality of settings creates a challenge for decentralization due to the different local practices and organizational policies that conflict [47]. Prior research has shown that people whose home base is distributed from their teammates, have different and even conflicting work practices influenced by organizational policies [66] [57], different reference frames of the technology (i.e. assumptions and expectations of use and purpose) [67] [75] [108], conflicting conventions for handling shared objects and groupware [37] [54] [89], different resources [89], and even use of different models and methodologies applied to the same problem though collaborating partners are of the same discipline and nominally of the same organization [56]. These differences are difficult to overcome and impact successful collaboration [65]. To our knowledge, no one has investigated how different perspectives, practices, and policies interact and impact decentralized work.

Relationship work: Decentralized activities distribute power throughout a social network. They are dynamic, flexible, and responsive to changing conditions. A key means by which people collaborate is through creating and sustaining human relationships [60]. Such "relationship work" in the contemporary context requires both technical capacity and social knowledge about the multiple stakeholders in the activity. Possible challenges for relationship work in decentralized activities are distributed relationships, rapid change, opacity of networks, cultural differences, varying motivations for participating in the shared activity, and time needed to build relationships.

Processes and coordination: Free/open source software development (FOSSD) is a widely practiced decentralized approach to building and sustaining large distributed software systems [11] [85]. It focuses on practicing the open access, examination, modification/creation, redistribution, and replication of shared, decentralized knowledge artifacts, development processes, and related socio-technical work practices. The artifacts include online chat transcripts, annotated source code, bug reports, etc. that act as boundary objects that can span multiple Web-based FOSSD projects [40]. The development processes include decentralized development or modification of a system's source code modules, and the collective composition, configuration, building, and testing of these modules into candidate or formal software system releases. The practices include self-organizing and continuously emerging FOSSD contributor roles and role migration paths [41], virtual project management (affecting FOSSD activity allocation, performance, oversight, and coordination) without formal manager roles or administrative authority [83], and others. Further, the OSSD project community relies on the decentralization of knowledge of the requirements and design of FOSS in order to insure commitment and socio-technical advancement with a project community [41] [85].

Privacy and awareness: When collaborative work is geographically and temporally distributed, collaborators find it challenging to be aware of each other's activities, routines, tasks, and availability. Yet, such awareness is crucial for increased efficiency and effectiveness of collaborative work [25] [39] [76] [63]. Collaborative software therefore increasingly provides means to disseminate awareness information to facilitate collaboration, such as in instant messaging systems, word processors [16], calendars [68], and programming environments [19]. Infrastructures that seamlessly and automatically capture, store, process and disseminate awareness have been implemented in a variety of domains such as workplaces [72], hospitals [10] [13], and conference centers [24].

However, collaborative needs for awareness are often at odds with individuals' desires for privacy. Prior studies [39] [69] [70] [71] [74] indicate that the inability to achieve a balance between awareness and privacy can lead to underuse of collaborative technology. Also, inadequate attention to privacy aspects may evoke strong user backlash, as was illustrated recently when the popular social networking site Facebook introduced new privacy-invasive awareness features. In such a case, organizations stand to lose their investment in collaborative technology, and face the prospect of longer-term damage due to the undermining of trust and credibility. It is therefore critical to consider privacy aspects when designing awareness mechanisms in collaborative systems. Patil & Kobsa [70] also found evidence that the motivation behind privacy concerns in collaborative settings is impression management, i.e. the desire of an individual to convey an impression of oneself appropriate for the context at hand.

Security and trust: Understanding the nature of decentralized applications and virtual worlds is critical before solutions can be designed for them. Our previous work [91] in building secure decentralized applications has revealed that developing appropriate threat and risk models is the most critical step in this process; it is only when threats can be characterized and are well-understood that effective countermeasures can be appropriately designed and deployed. Decentralization induces risk regarding the perceived multiple points of attack on information access or integrity from potentially insecure system interfaces. Any attempt to support coordination in an open decentralized system – one in which the set of participants may change and no central authority prevails to guarantee that all participants are non-malicious– must address, from the outset, the risks presented. The alternative is to create systems that will inevitably fall prey to malicious behavior and hence become insecure and vulnerable to exploits, surreptitious or remote take-over, or denial of service attacks.

Our previous work has revealed basic principles and effective techniques for designing specific types of decentralized systems that can (1) support security (information access) constraints at the architectural level [79], and (2) incorporate basic secure design and reputation-based trust models [3] [104] [77] [43] [21] in the architectures of participating members [93]. Since the participant architecture must protect itself against threats, appropriate countermeasures need to be incorporated within the architecture. Our experience with developing the PACE architectural style [94] for decentralized trust management gives us a good handle on how to identify countermeasures for threats.

Domain constraints can influence the choice of components in the participant architecture. For example, if the application requires exchange of rapidly changing critical data every few seconds, there may be a need for "estimator" components within the participant architecture to predict new data in case of communication delays [45]. Similarly, our previous work on software architecture-based decentralized trust management identified the need for a special component that encapsulates domain-specific conditions that determine trustworthiness [91]. If a participant architecture is composed of components belonging to different stake-holders, a connector-based security approach can be used to regulate information access within the participant architecture [79].

5. Research Partners

We have five research partners with whom we will be conducting our proposed study. The five are (a) The Aerospace Corporation, a federally funded research and development center (FFRDC), (b) Avaya Labs, a multi-site corporate laboratory focusing on research and development of telecommunications systems and software for business applications, (c) Discovery Science Center, a regional science center focusing on informal science education, (d) Northrop-Grumman Cyber Warfare Integration Center, a corporate laboratory focusing on development organizational knowledge management tools and techniques, and (e) Unimodal Inc., a startup venture focusing on research, development, and commercialization of personal rapid transit systems. Each of our five partners is engaged in the development and use of DVAS in their respective areas of interest. And each of these partners has agreed to provide our research team with access to their DVAS development or usage efforts, as well as inviting insights we gather that might further contribute to and improve the effectiveness of their development and usage efforts.

In the Computer Systems Research Department at *The Aerospace Corporation*, interest focuses on the development of an online environment for coordinating and providing oversight for a loosely coupled network of defense contractors who are building and integrating a system of systems application for the U.S. Air Force. These contractors are required to provide sensitive proprietary data and information regarding the performance of the systems they are developing or integrating to The Aerospace Corporation on behalf of the Air Force, yet these contractors do not want to have their proprietary data or information disclosed to their competitors. Thus, The Aerospace Corporation is interested in a new multi-contractor project oversight environment where they can securely receive and transmit proprietary data and information to/from the defense contractors they are coordinating, yet at the same time be able to partially reveal selected data or information about the performance of one or more artifacts to others in order to maximize the overall likelihood of success in the development. Additional interests within the scope of this management task include issue tracking and management, meetings, and software configuration management. Such a multi-contractor environment is envisioned as a DVAS.

In the Collaborative Applications Research group at *Avaya Labs*, our partner is focus on problems of how best to manage and coordinate distributed, multi-site, and multi-national software development projects through collaborative technologies. Though their software development projects are normally within their corporate boundaries, each site’s development team is located within a corporate profit center that may be operating in a different time zone, country, and work culture. They seek to develop a DVAS to help them visualize the location, availability, and technical skills/capabilities of software developers who are part of a multi-site software development projects, yet who might be reassigned or made unavailable to the project team during their development efforts. Participating software developers in turn seek ways to control how information about their status, expertise, and availability is shared or distributed with others on the team in different corporate locations.

At the *Discovery Science Center*, they are in the business of providing hands-on informal science education experiences to more than 120,000 K-8 grade students, and upwards of another 300,000 members of the public who visit DSC year round. DSC in collaboration with the UCI Game Culture and Technology Laboratory recently completed the development and deployment of a single-player and multi-player online science learning game environment called, *DinoQuest Online* [2]. DSC is now launching a new phase of development and internationalization of DQO that will provide support for collaborative problem solving among DQO game players (school age children) via social networking and communication services, as well as expanded multi-player game content, and multi-site networked game play across regional science centers in the U.S. and abroad. DQO was designed to conform to California Science Education Standards for the life sciences in grades K-6. As a result, DSC is providing Web-based access to DQO to students in more than 40 Southern California school districts, as well as developing teacher training and curricular materials that are designed to help teachers and students get the most of the DQO science learning game experience. But these teachers, students, parents, and school administrators are all interested in seeing how well such a game environment works for assisting or enabling young students to learn the basics of life science, and perhaps even plant the seed within them for a career in science. Members of each of these groups have expressed interest in being able to add to, extend, or modify the DQO science game content or software. This in turn has encouraged DSC to look for ways to accommodate open source software (and content) development methods as a way of further engaging its customer and partner base, but at the same time seeking to maintain or improve the quality and educational value of the DQO science learning games. Thus, the ongoing development and use of DQO is to make it more of a DVAS that both continues to support immersive science learning game play, as well as decentralized open source software development activities and local tailoring of DQO to regional science centers in the U.S. and elsewhere.

The Cyber Warfare Integration Laboratory at *Northrop Grumman* has been investigating potential applications of virtual world technologies to support the prototyping and development of advanced military systems and training applications. Their attention has been focused on articulating ways and means for system developers, designed, project managers, customers, and sub-contractors to better understand both what the envisioned system under development looks like, or how it is to be used/experienced, in order to better capture the various kinds of organizational knowledge and expertise that must be mobilized to insure the system's feasibility and subsequent production. They have been experimenting with the prototyping of new facilities and systems for military applications using virtual world technologies like *Second Life*, as well as a variety of computer game technologies. Thus, what makes their efforts relevant to our study is through how they employ, prototype, and evaluate different VAS technologies to engage and elicit project specific knowledge and expertise in ways they can capture, represent, and access across stakeholders working in different enterprises.

Unimodal Inc. is a start-up venture focusing on the development and commercialization of personal rapid transit (PRT) systems for regional and venue-specific deployments. PRT systems are still mostly experimental, yet there is great interest in their potential to relieve transportation congestion and reduce the carbon footprint of current transit options. But what do such systems look like, how do they operate, where/how will they be installed and configured to run, are among the multitude of questions that consumers, system developers, financial investors, and regional transportation and governmental authorities are asking. To help answer such questions, Unimodal has engaged a research team lead by Prof. Crista Lopes to create PRT system mock-ups within the virtual world of Second Life to better help articulate the answers to these questions, but also to help identify and answer new questions that emerge along the way. In this regard, the PRT system research and development effort underway at Unimodal is employing virtual world technologies to help a diverse community of stakeholders to see/visualize and interactively experience a virtual PRT system operating in a virtual world, in order to help these stakeholders make sense of what they see and experience, as well as to help elicit their concerns, interests, and questions regarding the emerging design and commercialization of a proposed PRT system at different installation sites. As such, it is yet another variation of a unique DVAS development and usage effort for us to study and potentially influence in the future.

6. Research approach and methods

Our research approach is empirical. Our methods are primarily focused on qualitative observation and inquiry through field study, and experimental in that they focus on prototyping and evaluating new DVAS concepts, techniques, and tools in the context of different types of organizations and applications of specific DVAS development and usage efforts.

It is always a challenge to model the array of factors that affect real world decentralized work activities, e.g. work pressures, career trajectories, local influences, routines, experience and expertise, etc. Adding another layer of variability arises through a consideration of how such factors are mediated through interaction or collaborative activities that take place in virtual settings. Therefore, we find the best methodology for understanding the complex interplay of organizational, social, and technological factors involved in reconciling alternative activity system configurations (whether decentralized, virtualized, or both together) is through *in situ* studies. However, this alone cannot produce generalizable results unless effort is made to structure and situate these studies across a set of diverse approaches to developing and using DVASs that are found in a diverse set of organizational settings. Therefore, our field studies must also produce data, activity patterns, work/play practices, development processes, and usage scenarios/experiences that can be analytically coded and compared. It is through comparative analysis across cases arising in different settings, with different DVAS configurations or usage scenarios, or with new concept demonstration or prototyped DVAS mechanisms that we can articulate more generalizable results. Here's how we make this work.

First, each of our six analytical variables (described in Section 3) is associated with one project investigator who is already vested in prior research, and therefore brings both prior research expertise as well as an analytical eye with which to observe conditions and events that impinge on their variable under study.

Second, we have identified five research partners with who we will engage our field studies. However, each of our partners presents a different set of research variables of interest. Table 1 represents our current understanding as to which research variables are central to the DVAS development or use efforts underway at each of our five research partners.

Table 1: Initial variable-partner matches

	Aerospace Corporation	**Avaya Labs**	**Discovery Science Center**	**Northrop Grumman**	**Unimodal Inc.**
Representations and realities			XXX	XXX	XXX
Conflicting policies and practices	XXX	XXX		XXX	
Relationship work		XXX	XXX	XXX	XXX
Processes and coordination	XXX	XXX	XXX	XXX	XXX
Privacy and awareness	XXX	XXX			
Security and trust	XXX		XXX		XXX

This distribution of research variable-partner matches signifies that we have a sufficient diversity of variable mixes across the five research partner sites. This provides us a basis for conducting comparative analyses across (a) individual variable-partner matches (comparing results from studies of each cell in the table), (b) range of variables found in each organization (comparing across a column), (c) range of organizations articulating each variable (comparing across each row), and (d) all research variables across all organizations (comparing across columns and rows together). This gives us a maximum comparative analytical capability that can inform or suggest generalizations that account for data that articulate the variables of our proposed study.

Last, our research method must insure a balanced study of socio-technical processes, practices, constraints, opportunities, and affordances that help characterize how different DVASs are developed and used in their particular multi-organizational setting. Accordingly, three of our research variables (security and trust, privacy and awareness, and representations and realities in an immersive world) are associated with experimental development and usage of new DVAS technologies--tool or functional mechanism prototypes that may be incorporated into their respective research partner DVAS project effort. The other three research variables (processes and coordination, relationship work, and conflict practices and policies) focus on identifying, articulating, and modeling/codifying recurring social practices, activity patterns, and processes that characterize the development and/or use of situated DVAS. Though we will begin focusing on one enterprise partner for each analytic variable, we will continue studying these variables with other partners as well as shown in Table 1.

For our study of social practices and processes we will apply rigorous qualitative methodology as outlined by [33] [102] [103] [28] [27]. Methods will consist of intensive observation of work activities, repeated semi-structured interviews among participants (by telephone or in-person), and document analysis, when documents are available. The use of these methods provides a rich corpus of data for understanding the complex processes and relationships of interest to us.

Representations and realities: We will study prototyping of immersive models and simulations of personal rapid transit (PRT) systems within a virtual world of *Second Life*. This will include ongoing work with our research partner Unimodal Inc. and several county governments in Southern California, in order to expand the scope and diversity of issues and public policy concerns that are being addressed through early use of the current PRT simulation in *Second Life*. We will make the PRT simulation world open and publicly available for experimental use and in-world evaluation by the diverse participants, and will engage with the local community in having people experience the system.

Conflicting practices and policies: Ethnographic techniques will be employed with Northrup Grumman to 1) understand the array of different perspectives and practices that exist in DVAS found in and 2) to investigate the impacts that new DVAS development and usage that emerge. Ethnographic methods have been widely deployed in the study of collaborative distributed organizational settings yielding insightful findings (e.g., [66] [80] [4] [65] [31] [31] [54] [61]). In some cases, it will be beneficial to isolate particular variables of interest in a laboratory setting for usability testing or to test the effectiveness of particular features of a system. For these investigations, experimental studies in a laboratory will be conducted.

Relationship work: Building on previous research on relationship work [60] [61] [62], we will examine relationship work in decentralized activities that arise in the development and use of virtual worlds for organizational knowledge management at Northrop Grumman. Qualitative work including field observations, in-depth interviews, and document analysis will inform the design of focused surveys to gain broader insights and identify quantitative patterns about how people create and sustain relationships in decentralized activities. These insights will be used to articulate the requirements and design features of DVAS technologies intended to support relationship work in decentralized settings.

Processes and coordination: The primary approach involves empirical study of DVAS development and deployment projects via multi-modal ethnographic discovery and modeling of observed development processes, work practices, and project community dynamics [84] found at our research partners, such as the Discovery Science Center. Current research studies [85] employ this approach in the study of open source software development projects focusing on the development of (a) Internet information infrastructure systems; (b) networked computer games, (c) scientific research in astrophysics and bioinformatics, and (d) administrative computing applications. In the proposed effort, we see DVAS development projects underway or under consideration in each of our enterprise partners, including in the Discovery Science Center in its efforts to continue its development and expansion of interactive, online science learning games and collaborative problem-solving environments publicly accessible over the Web.

Privacy and awareness: We plan to conduct a field study of distributed collaborators at one of our industry partners, Avaya Labs, to uncover attitudes and practices regarding impression management and how these relate to privacy considerations. Data gathering will be performed through surveys, interviews, non-participant observation, focus groups, and experiments. Multiple methodologies are needed partly to gain a holistic understanding of impression management practices, and partly to offset the methodological challenges in privacy research [73].

Results of the field study will be used to generate requirements for prototype implementations of new impression management and privacy controls available at the user interface to Avaya Labs' DVAS. To allow early exploration, the initial prototype will be restricted to an electronic communication system in use at Avaya Labs, such as an instant messaging system. Based on the results, we will extend it to provide a generic and extensible solution for other collaborative systems (e.g., shared calendars, source code repositories, email, blogs). It is likely that such a solution will take the form of a middleware layer that provides impression visibility and management as a "service" via an Application Programming Interface (API). Alternatively, it may take the form of a suite of plug-ins for individual collaborative systems.

Security and trust: Our approach uses threat and risk models that apply to DVAS to identify/develop (a) appropriate trust models to counter them, *and* (b) secure technologies that can be encapsulated within the participant architectures to support the trust models and hence counter the threats. Our initial specific focus will be to work with our partner at The Aerospace Corporation to investigate the threats, risks, and requirements for trust that arise in the context of their effort to develop and project oversight environment for managing system of systems development projects.

Identifying the threats and risks is our first task. Our previous work [95] has already identified a number of critical threats of decentralized systems. But virtual worlds and interactions between them will surely impose some additional constraints and risks that need to be understood. Therefore, we plan to pursue an in-depth examination of the experiences and the findings of the virtual world research community, the CSCW community [42] [97] [90] [7] and our own prior experience to pinpoint the dominant characteristics and vulnerabilities of these systems.

Second, we will identify appropriate trust models. Numerous trust models exist in the research literature [34] [106]. Depending upon the nature of the collaboration involved, a suitable trust model that can counter the identified threats needs to be identified. For example, if the entities are mostly concerned with regulating access to critical resources, credential and policy-based trust models [14] [15] [52] [101] may be adopted. Or, if entities need to determine whether reported information can be trusted, reputation-based trust models [105] [44] [38] [81] [99] [58] [92] may be adopted. Both kinds of models may be required in some circumstances. We plan to develop techniques and tools to support the selection of appropriate trust models. Towards this goal, we have already developed an initial suite of tools that help an application developer to choose a suitable reputation-based trust model. These include the TREF framework [95] that helps identify an initial set of reputation models and the SIFT simulator [96] that simulates this set of reputation models under varying threat scenarios.

Third, we will identify design principles that will provide specific guidance on how to design secure participants. For this, we will draw upon our prior experiences [91] [79] as well as leverage contemporary literature on various types of trust and security frameworks [35] [17] [51] [18] [26] [36]. The final step in our approach is to leverage our extensive experience with software architectural styles to choose an appropriate architectural style to incorporate these design guidelines.

7. Assessment methods and metrics

In order to assess the impact of the technologies we develop, we will first conduct a baseline assessment of the six variables of interest at each research site. This work can be conducted by graduate students guided by the project faculty investigators and will involve interviewing and observing our research partners to understand current practices and to understand what is important to them in terms of enhanced socio-technical support. Thus our assessment measures will emerge from the research and our interactions with our partners. This means that we will have a set of measures that include both measures common to all partners as well as some that are specific. As our sites are carefully chosen to represent diverse DVAS, we believe this approach will be fruitful in leading to scientific understanding of DVAS as a single class of systems as well as understandings about specific kinds of DVAS. Once the baseline study has been conducted we will be in a position to measure the impact of the technologies and practices we design on the six variables, to ascertain how they change activities in DVAS.

Beyond this, how will we or others come to know whether the proposed research undertaking has produced meaningful and usable results from our study? To help determine this, we presented above a representative set of research questions aligned with each of the six research variables that are the focus of our study. The answers, explanations, interpretations, models, or metrics we provide in response to the systematic observational data we collect, and the multi-layered comparative analysis we undertake, ultimately determine whether we have realized our goals. As such, we now describe in greater detail how we plan to assess our research goals in each of the six research variable areas.

Representations and realities: Building on our efforts to date in prototyping an immersive virtual world for evaluating a new PRT system with our research partner, Unimodal Inc., we will collect and interpret data addressing the following detailed variables of interest. These include determining how many people visit the SL site, what stakeholder communities they represent, and the degree to which the PRT immersive simulation is used/referenced in stakeholders meetings. The particular questions of blending real and virtual elements will focus on identifying the critical circumstances and attributes when melding the elements is successful (such as when an element is itself virtual, such as control software). Issues of representation and reality will similarly be addressed in the context of our other partners.

Conflicting policies and practices: Following the approach used by Mark [54], we will focus on identifying what new conventions of practice and policy have been formed, how they address decentralized work, and importantly, whether they are *followed* by collaborating partners. Perspectives and practices, and policies are dynamic, and constantly evolve and we will also track the extent to which these change over time to adapt to new decentralized configurations. Ethnographic techniques will enable us to identify and evaluate the relevance of new practices and policies to conditions.

Relationship work: The research will examine the entire ecology of tools and techniques for relationship work used by participants, beginning at our research partner, Northrop Grumman. We will investigate how their tools and techniques are used in conjunction with existing collaboration tools such as those for social networking tools, email, instant messaging, video conferencing, and so on. We will identify ways in which the tools enhance one another as well as remaining gaps that need to be filled for more effective relationship work. We will investigate how new mixes of tools emerge, some replacing, enhancing, or complementing others. We will assess how work practices change, and in what ways, using the new tools, discovering how work practices evolve to meet the demands of relationship work in decentralized activities. Subsequent studies will pursue these questions in the context of the other research partners, several of whom have similar situations.

Processes and coordination: Our approach to the use of multi-modal ethnographic and modeling of DVAS processes, practices, and project dynamics that we observe in our research partners is founded on principles of grounded theory development via ongoing comparative analysis of multiple cases of situated practice. Our data collection and analysis methods for process discovery are geared towards both formative and summative assessments of what we can learn from our studies. Our formative assessments rely on a transparent, reconstructable method for data collection, coding and cross-coding, and comparative analysis of multiple cases (both similar and dissimilar). Our summative assessments result from our multi-modal analysis and modeling of the data we collect, and the alternative interpretations we use to present such summary findings and models [84] [85].

Privacy and awareness: We will employ several evaluation techniques throughout the prototype building activities, specifically rapid prototyping and heuristic evaluation in the context of our study at Avaya Labs. We will also conduct usability studies at intermediate stages to iteratively improve the prototype. We will deploy the final prototype versions at UC Irvine as well as industry partners, and collect usage logs as well as conduct interviews with users to understand how well the prototypes meet their goal of supporting impression visibility and management (and in turn of improving privacy management). We will also gather feedback for further improvements. Cross-site comparisons will be made to illuminate how the system should adjust to different organizational and work contexts.

Security and trust: We will evaluate our approach in the context of our research partners, beginning with The Aerospace Corporation's system of systems project oversight and coordination situation. Specifically, we will begin with study of their problem domain, identifying the threats and risks present. We will then use previously developed tools and techniques [93-95] as the basis for identifying a suitable trust model (for those threats so amenable). Such trust models will then be integrated within each participant's architecture using an architectural style that leverages the past PACE work and incorporates connector-based security policies [79] [78]. Next, the application comprising of these participant architectures will be subjected to a variety of threat scenarios and the behavior of each participant studied. These experiments will help us evaluate whether and to what extent these participant architectures in concert with the trust model allow DVAS participants to establish useful trust relationships with each other and control correctly the security of information being exchanged. The threats relevant to this situation are primarily those of inappropriate data access, rather than overtly malicious behavior. Iteration on this process is expected, as shortcomings in the prior work are identified relative to the new circumstances of DVAS, and as differences between our different partner sites are discovered.

8. Intellectual merit and broader impacts

Our focus on DVASs provides the opportunity to study people's experiences with simulated versions of new technological systems while these systems are being conceived and designed, prior to their eventual deployment and use in a broader social setting. We believe that research on DVASs will have a vital impact on society. As development and use of DVASs becomes more common practice and as organizations continue to become more decentralized, new methods and policies will need to be identified and tested to enable people to collaborate successfully. We also believe that our study has economic value as it will help organizations to carry out decentralized work effectively with smoother coordination, so that they can better compete in the global market. DVASs will be resilient to environmental disruptions as collaboration will be able to be conducted from anywhere, anytime, using representations of people, artifacts, and activities. Our results will also have an important impact on higher education, as new people entering the workforce will have to gain skills in developing systems, and in conducting work, in a decentralized setting. Our management plan (below) provides more details on how our results will be integrated into different education settings.

Understanding the unique challenges of relationship work in decentralized activities and designing DVAS technologies to meet the challenges will have both theoretical and practical impact. Theoretically we want to determine the contours of decentralized activity, allowing comparison to other social forms such as traditional hierarchies, communities of practice [100] and the rational bureaucracies described by Max Weber. We believe this theoretical work will be broadly useful in the fields of human-computer interaction, computer-supported collaborative work, organizational studies, and design research and practice that incorporate the wider social context in which technologies are developed and used. Practically, our DVAS studies and prototype tools, techniques, and concepts will enable more effective relationship work in decentralized activities. Through study of their use we will be able to understand future directions needed for continued development of tools for decentralized activity.

Our effort to develop empirically grounded models and theories of decentralized virtual environment development processes, work practices, and project community dynamics both builds on and complements our current studies of open source software development projects. Our studies of the development and deployment of DVASs with our enterprise partners will serve as reference models for how organizations can learn practices, guidelines, models, and theories to follow. For example, our partnership with the Discovery Science Center will provide valuable results to other similar learning centers when they seek to develop and deploy online interactive exhibits and learning games that foster education practices. Similarly, results of our studies with our other enterprise partners can be applied to similar organizations.

The long-term impact of our research on privacy and awareness in the development and use of DVAS is likely to be substantial. Achieving our outcomes will lead privacy-sensitivity to become a standard design requirement in the development of DVASs and similar collaborative systems, and mechanisms for the user to manage their impression on others will become regular to-do items in software design specifications. This is especially important in the light of the fact that most collaborative systems so far focus on the awareness benefits since those are the primary purpose behind building the system, while privacy management often gets secondary attention. We expect some impacts may even go beyond collaborative systems and affect all systems that involve privacy and impression management by computer users.

Our proposed approach, techniques, and tools to the study of how best to provide effective security and trust mechanisms when developing a DVAS will contribute towards the development of secure collaborative open decentralized systems that will be acceptable to and usable by the target audiences. The importance of this should not be underestimated: if appealing DVASs are created, and if they remain defenseless in the face of malicious activities, they will be unused. Provision for security and trust is essential, not an option. Our approach, moreover, will not just identify abstract policies and practices to follow, but provide specific techniques for incorporating security and trust technologies in DVAS software.

9. Results from Prior NSF Funded Research

Project investigators Scacchi, Kobsa, Lopes, Mark, Redmiles, and Taylor all have prior NSF support from at least three different NSF funded research grants.

Award IIS-0205724, amount: $1,800,000, 7/2002-9/2008 (Completed). "ITR: An Integrated Social and Technical Approach to the Development of Distributed, Inter-Organizational Applications." PI/Co-PIs: Taylor, Richardson, Kobsa, Redmiles, Dourish, Mark, van der Hoek. Senior Personnel: Scacchi. This project was a broad empirical study examining the relationship between distributed, inter-organizational management structure and the structure of software built by and for those inter-organizations. The investigation included (a) examining decentralized software development projects, both open source and more standard approaches, (b) examining the role of privacy in decentralized systems, (c) examining security and trust concerns in decentralized systems, and (d) creating technologies to support decentralized development. Over 130 publications resulted from this project, including books, journals, conference papers, and workshops. Over 20 Ph.D students were supported over the course of the project.

Issues of decentralization and consensus that occur in open-source communities were explored, where diverse agencies with independent, often conflicting goals are yet able to come together to produce a high-quality product. Interactions between different teams at NASA were studied, whose communication and interaction patterns exhibit characteristics of decentralization despite ostensibly being part of the same agency. Investigators examined how workers negotiate and manage their membership in multiple communities within a large-scale organization. Investigations delved into how people manage to negotiate and switch their identity/membership among these multiple communities, and why people need these different communities to function effectively in the organization.

To support awareness in distributed configuration management and to aid coordination and collaboration activities among geographically distributed developers, several software tools and visualization techniques were developed as part of this project. At the same time, this need of awareness is frequently at odds with an individual's desire to keep private some of this information. Effectively balancing awareness and privacy needs has proven to be a significant challenge for designers of awareness systems and related infrastructures. Investigation also focused on identifying concerns arising out of decentralization that affect security and trust management solutions for decentralized systems, addressing those concerns through the development of a suitable architectural style, providing design guidelines towards constructing decentralized systems, constructing actual systems using that style, evaluating these systems against threat scenarios, and using experimental results to refine architectural style and design guidelines for future use.

Award no. IIS-0534771, Amount: $335,000, 11/01/05 to 10/31/08, *Discovering the Processes, Practices, Community Dynamics and Principles for Developing Open Source Software Systems*, PI: Walt Scacchi. This research seeks to account for prior results and recent research findings in the area of free/open source software development processes, work practices, and project dynamics. This effort address topics such as (a) the identification of software informalisms as different types of artifacts and communication media that are used to facilitate and coordinate FOSS development projects, as well as serve as decentralized sources of developer knowledge [29] [85], and (b) the role migration and socio-cultural mobilization of socio-technical resources, mitigating conflict, and facilitating career/occupational development in FOSS projects [30] [41].

Award no. 0724806, Amount: $726,455, 09/01/07 to 08/31/10, *SDCI Data New: Trust Management for Open Collaborative Information Repositories: The CalSWIM Cyberinfrastructure*, PI: Cristina Lopes. This project will study and support the California Sustainable Watershed Information Manager. The work just started aims at creating a publicly updatable, Wiki-based online encyclopedia of *all things watershed* that includes all watersheds in California. The CalSWIM Wiki will only succeed if issues of trust are taken into consideration in its design. Professor Lopes also has prior results from NSF supported projects including a CAREER award, in which she continued her original innovative work on Aspect-Oriented Programming [46] with an emphasis towards assessment of new approaches to software design and development.

References

[1] Gartner Group. <http://www.gartner.com/it/page.jsp?id=503861>, 2007.

[2] DinoQuest Online. <http://www.dinoquestonline.org>, 2007.

[3] Abdul-Rahman, A. and Hailes, S. A Distributed Trust Model. In *Proceedings of the New Security Paradigms Workshop.* Langdale, Cumbria UK, 1997.

[4] Ackerman, M.S., Hindus, D., et al. Hanging on the Wire: A Field Study of an Audio-Only Media Space. *ACM Transactions on Computer-Human Interaction.* 4(1), p. 39-66, 1997.

[5] Ahuja, M.K. and Carley, K. Network Structure in Virtual Organizations. *Organization Science.* 10(6), p. 741-757, 1999.

[6] Alter, S. The Work System Method: Connecting People, Processes, and IT for Business Results. Work Systems Press, 2006.

[7] Appelt, W., Pajo, S., et al. Supporting Trust Building in Distributed Groups by Appropriate Security Technology. In *Proceedings of GROUP '07.* Sanibel Island, Florida, USA, November 4-7, 2007.

[8] Bainbridge, W.S. Review: The Scientific Research Potential of Virtual Worlds. *Science.* 317(5837), p. 472-476, July, 2007.

[9] Baker, W. The Network Organization in Theory and Practice. In *Networks and Organizations,* Nohria, N. and Eccles, R. eds. Harvard Business School Press: Boston, MA, 1992.

[10] Bardram, J.E. Applications of Context-Aware Computing in Hospital Work: Examples and Design Principles. In *Proceedings of the 2004 ACM Symposium on Applied Computing.* ACM Press, New York, NY, USA. Nicosia, Cyprus, 2004.

[11] Benkler, Y. The Wealth of Networks: How Social Production Transformation Markets and Freedom. Yale University Press, 2006.

[12] Bergman, M., Lyytinen, K., et al. Boundary Objects in Design: An Ecological View of Design Artifacts. *Journal of the Association for Information Systems, Special Issue on Systems Analysis and Design.* Forthcoming.

[13] Black, J.P., Segmuller, W., et al. Pervasive Computing in Health Care: Smart Spaces and Enterprise Information Systems. In *Proceedings of the Second International Conference on Mobile Systems, Applications, and Services.* Boston, MA, 2004.

[14] Blaze, M., Feigenbaum, J., et al. Decentralized Trust Management. In *Proceedings of the IEEE Symposium on Security and Privacy.* p. 164-173, May, 1996.

[15] Blaze, M., Feigenbaum, J., et al. RFC 2704 - The KeyNote Trust-Management System version 2. <http://www.faqs.org/rfcs/rfc2704.html>, 1999.

[16] Cadiz, J.J., Gupta, A., et al. Using Web Annotations for Asynchronous Collaboration Around Documents. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW 2000).* ACM Press, New York, NY. Philadelphia, PA, USA, 2000.

[17] Cahill, V., Gray, E., et al. Using Trust for Secure Collaboration in Uncertain Environments. *IEEE Pervasive Computing Mobile and Ubiquitous Computing.* 2(3), p. 52-61, August, 2003.

[18] Capra, L. Engineering Human Trust in Mobile System Collaborations. In *Proceedings of the 12th International Symposium on the Foundations of Software Engineering (SIGSOFT 2004/FSE-12).* Newport Beach, California, USA, November, 2004.

[19] Cheng, L.-T., Hupfer, S., et al. Jazzing up Eclipse with Collaborative Tools. In *Proceedings of the 2003 OOPSLA Workshop on Eclipse Technology eXchange.* ACM Press, New York, NY, USA. Anaheim, CA, USA, 2003.

[20] Curtis, P. MUDing: Social Phenomena in Text-Based Virtual Realities. In *Proceedings of the Conference on Directions and Implications of Advanced Computing.* 1992.

[21] Damiani, E., di Vimercati, S.D.C., et al. A Reputation-Based Approach for Choosing Reliable Resources in Peer-to-Peer Networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security.* Washington DC, November, 2002.

[22] DeSanctis, G. and Jackson, B. Coordination of Information Technology Management: Team-based Structures and Communication Systems. *Management Information Systems.* 10(4), p. 85-110, 1994.

[23] DeSanctis, G. and Monge, P. Introduction to the special issue: Communication Processes for Virtual Organizations. *Organization Science.* 10(6), p. 693-703, 1999.

[24] Dey, A.K., Salber, D., et al. The Conference Assistant: Combining Context-Awareness with Wearable Computing. In *Proceedings of the 3rd IEEE International Symposium on Wearable Computers.* IEEE Computer Society, Washington, DC, USA. 1999.

[25] Dourish, P. and Bellotti, V. Awareness and Coordination in Shared Workspaces. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW'96).* ACM. Toronto, Ontario, 1996.

[26] Dragovic, B., Hand, S., et al. Managing Trust and Reputation in the XenoServer Open Platform. In *Proceedings of the First International Conference on Trust Management.* Crete, Greece, May 28-30, 2003.

[27] Dubé, L. and Paré, G. Rigor in Information Systems Positivist Case Research: Current practices, trends, and recommendations. *MIS Quarterly.* 27(3), p. 597-635, 2003.

[28] Eisenhardt, K.M. Building Theories from Case Study Research. *Academy of Management Review.* 14(4), p. 532-550, 1989.

[29] Elliott, M., Ackerman, M.S., et al. Knowledge Work Artifacts: Kernel Cousins for Free/Open Source Software Development. In *Proceedings of the ACM Conference on Support of Group Work (GROUP07).* p. 177-186, Sanibel Island, FL, 2007.

[30] Elliott, M. and Scacchi, W. Mobilization of Software Developers: The Free Software Movement. *Information, Technology and People.* To appear.

[31] Feldman, M.S. Resources in Emerging Structures and Processes of Change. *Organization Science.* 15(3), p. 295-309, May-June, 2004.

[32] Gay, J. ed. Free Software Free Society: Selected Essays of Richard M. Stallman. GNU Press, Free Software Foundation: Boston, MA, 2002.

[33] Glaser, B.a.S., A. The Discovery of Grounded Theory: Strategies for Qualitative Research. Aldine de Gruyter: New York, 1967.

[34] Grandison, T. and Sloman, M. A Survey Of Trust in Internet Applications. *IEEE Communications Surveys.* 3(4), December, 2000.

[35] Grandison, T. and Sloman, M. Trust Management Tools for Internet Applications. In *Proceedings of the 1st International Conference on Trust Management.* Crete, Greece, May, 2003.

[36] Gray, E., O'Connell, P., et al. Towards a Framework for Assessing Trust-Based Admission Control in Collaborative Ad Hoc Applications. Distributed Systems Group, Department of Computer Science, Trinity College, Report TCD-CS-2002-66, 2002.

[37] Grudin, J. Why CSCW Applications Fail: Problems in the Design and Evaluation of Organization of Organizational Interfaces. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW '88).* p. 85-93, ACM Press. Portland, Oregon, 1988.

[38] Gupta, M., Judge, P., et al. A Reputation System for Peer-to-Peer Networks. In *Proceedings of the Thirteenth ACM International Workshop on Network and Operating Systems Support for Digital Audio and Video.* Monterey, California, June 1-3, 2003.

[39] Hudson, S.E. and Smith, I. Techniques for Addressing Fundamental Privacy and Disruption Tradeoffs in Awareness Support Systems. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW '96).* p. 248-257, ACM Press, New York, NY, USA. Boston, MA, 1996.

[40] Jensen, C. and Scacchi, W. Process Modeling Across the Web Information Infrastructure. *Software Process--Improvement and Practice.* 10(3), p. 255-272, July-September, 2005.

[41] Jensen, C. and Scacchi, W. Role Migration and Advancement Processes in OSSD Projects: A Comparative Case Study. In *Proceedings of the 29th International Conference on Software Engineering (ICSE 2007).* p. 364-374, Minneapolis, MN, May, 2007.

[42] Jones, S. and Marsh, S. Human-Computer-Human Interaction: Trust in CSCW. *SIGCHI Bulletin.*

29(3), 1997.

[43] Josang, A. and Ismail, R. The Beta Reputation System. In *Proceedings of the 15th Bled Electronic Commerce Conference.* Bled, Slovenia, June 17-19, 2002.

[44] Kamvar, S., Schlosser, M., et al. The EigenTrust Algorithm for Reputation Management in P2P Networks. In *Proceedings of the Twelfth International WWW Conference.* Budapest, Hungary, May 20-24, 2003.

[45] Khare, R. and Taylor, R.N. Extending the REpresentational State Transfer Architectural Style for Decentralized Systems. In *Proceedings of the 26th International Conference on Software Engineering (ICSE 2004).* Edinburgh, Scotland, UK, May, 2004.

[46] Kiczales, G., Lamping, J., et al. Aspect-Oriented Programming. In *Proceedings of the 11th European Conference on Object-Oriented Programming.* Volume 1241 of LNCS, p. 220-42, Springer-Verlag. 1997.

[47] Kling, R. Social Analyses of Computing: Theoretical Perspectives in Recent Empirical Research. *ACM Computing Surveys.* 12(1), p. 61-110, 1980.

[48] Kling, R. and Scacchi, W. The Web of Computing: Computing Technology as Social Organization. In *Advances in Computers*, Yovits, M. ed. 21, p. 3-85, 1982.

[49] Kock, N. Benefits from Virtual Organizations from Distributed Groups. *Communications of the ACM.* 43(11), p. 107-112, 2000.

[50] Lee, C., Dourish, P., et al. The Human Infrastructure of Cyberinfrastructure. In *Proceedings of the ACM Conference on CSCW (CSCW'06).* p. 483-492, Banff, Alberta, Canada, 2006. Nominated for Best Paper.

[51] Lee, S., Sherwood, R., et al. Cooperative Peer Groups in NICE. In *Proceedings of the IEEE Infocom.* San Francisco, USA, April 1-3, 2003.

[52] Li, N., Mitchell, J.C., et al. Design of a Role-Based Trust-Management Framework. In *Proceedings of the 2002 IEEE Symposium on Security and Privacy.* p. 104-120, 2002.

[53] Lopes, C., Kan, L., et al. PRT Simulation in an Immersive Virtual World. In *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks, and Systems (SIMUTools).* In submission.

[54] Mark, G. Conventions and Commitments in CSCW Distributed Groups. *Computer-Supported Cooperative Work: The Journal of Collaborative Computing.* 11(3-4), p. 349-387, 2002.

[55] Mark, G., Gonzalez, V., et al. Reconciling Different Perspectives: An Experiment on Technology Support for Articulation. In *Proceedings of the Fifth International Conference on the Design of Cooperative Systems (COOP 2002).* p. 23-37, IOS Press. Saint-Raphaël, France, June 4-7, 2002.

[56] Mark, G., Abrams, S., et al. Group-to-Group Distance Collaboration: Examining the "Space Between". In *Proceedings of the 8th European Conference of Computer-Supported Cooperative Work (ECSCW '03).* p. 99-118, Helsinki, Finland, September 14-18, 2003.

[57] Markus, L. Power, Politics, and MIS Implementation. *Communications of the ACM.* 26(6), p. 430-444, 1983.

[58] Marti, S. and Garcia-Molina, H. Limited Reputation Sharing in P2P Systems. In *Proceedings of the ACM Conference on Electronic Commerce.* New York, USA, May 17-20, 2004.

[59] Morla, R. and Lopes, C. Blurring the Line between Simulation and Real-world Prototypes using Second Life™. In 9th International Conference on Ubiquitous Computing (UbiComp 2007) - poster: Tyrol, Austria, 2007.

[60] Nardi, B., Whittaker, S., et al. ContactMap: Integrating Communication and Information through Visualizing Personal Social Networks. *Communications of the Association for Computing Machinery.* April, p. 89-95, 2002.

[61] Nardi, B., Whittaker, S., et al. NetWORKers and Their Activity in Intensional Networks. *The Journal of Computer-Supported Cooperative Work.* 11(1-2), p. 205-242, 2002.

[62] Nardi, B.A. Beyond Bandwidth: Dimensions of Connection in Interpersonal Communication. *Journal of Computer Supported Cooperative Work.* 14(2), p. 91-130, 2005.

[63] Neale, D.C., Carroll, J.M., et al. Evaluating Computer-Supported Cooperative Work: Models and

Frameworks. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work.* ACM Press, New York, NY. Chicago, IL, 2004.

[64] Noll, J. and Scacchi, W. Supporting Software Development in Virtual Enterprises. *JoDI - Journal of Digital Information.* 1(4), 1999.

[65] Olson, G. and Olson, J. Distance Matters. *Human-Computer Interaction.* 15(2/3), p. 139-178, 2000.

[66] Orlikowski, W.J. Learning from Notes: Organizational Issues in Groupware Implementation. *The Information Society.* 9(3), p. 237-250, 1993.

[67] Orlikowski, W.J. and Gash, D.C. Technological Frames: Making Sense of Information Technology in Organizations. *ACM Transactions on Information Systems.* 12(2), p. 174-207, 1994.

[68] Palen, L. Social, Individual and Technological Issues for Groupware Calendar Systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI).* ACM Press, New York, NY, USA. Pittsburgh, PA, USA, 1999.

[69] Patil, S. and Kobsa, A. Instant Messaging and Privacy. In *Proceedings of the HCI 2004 - The 18th British HCI Group Annual Conference.* p. 85-88, Leeds, England, September, 2004.

[70] Patil, S. and Kobsa, A. Privacy in Collaboration: Managing Impression. In *Proceedings of the First International Conference on Online Communities and Social Computing.* Las Vegas, NV, 2005.

[71] Patil, S. and Kobsa, A. Uncovering Privacy Attitudes in Instant Messaging. In *Proceedings of the 5th ACM Conference on Supporting Group Work (Group '05).* p. 101-104, ACM Press, New York, NY. Sanibel Island, FL, 2005.

[72] Patil, S. and Lai, J. Who Gets to Know What When: Configuring Privacy Permissions in an Awareness Application. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI).* ACM Press, New York, NY. Portland, OR, 2005.

[73] Patil, S., Romero, N., et al. Privacy and HCI: Methodologies for Studying Privacy Issues. In *Proceedings of the ACM Conference on Human Factors in Computing Systems.* ACM Press, New York, NY. Montréal, Québec, Canada, 2006.

[74] Patil, S. and Kobsa, A. Privacy Considerations in Awareness Systems: Designing with Privacy in Mind. In *Awareness Systems: Advances in Theory, Methodology and Design,* Markopoulos, P., de Ruyter, B., and Mackay, W. eds. Springer Verlag: Berlin, Heidelberg, New York, Forthcoming.

[75] Pinch, T. and Bijker, W. The Social Construction of Facts and Artifacts. In *The Social Construction of Technological Systems.* p. 17-50, MIT Press: Cambridge, MA, 1987.

[76] Prinz, W., Mark, G., et al. Designing Groupware for Congruency in Use. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW '98).* p. 373-382, ACM Press. Seattle, WA, November, 1998.

[77] Pujol, J., Sanguesa, R., et al. Extracting reputation in multi agent systems by means of social network topology. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems.* Bologna, Italy, July 15-19, 2002.

[78] Ren, J., Taylor, R., et al. Towards An Architectural Treatment of Software Security: A Connector-Centric Approach. In P*roceedings of the Workshop on Software Engineering for Secure Systems (SESS 05),* held in conjunction with ICSE 2005. St. Louis, MO, May, 2005.

[79] Ren, J. A Connector-Centric Approach to Architectural Access Control. Thesis. Information and Computer Science, University of California, Irvine, p. 222, 2006. <http://www.ics.uci.edu/~jie/Thesis.pdf>.

[80] Rogers, Y. Exploring Obstacles: Integrating CSCW in Evolving Organizations. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW '94).* p. 67-78, 1994.

[81] Sabater, J. and Sierra, C. REGRET: A Reputation Model for Gregarious Societies. In *Proceedings of the 4th Workshop on Deception, Fraud and Trust in Agetn Societies.* Montreal, Canada, 2001.

[82] Scacchi, W. Understanding the Requirements for Developing Open Source Software Systems. *IEE Proceedings--Software.* 149(1), p. 24-39, February, 2002.

[83] Scacchi, W. Free/Open Source Software Development Practices in the Computer Game Community. *IEEE Software.* 21(1), p. 59-67, January/February, 2004.

[84] Scacchi, W., Jensen, C., et al. Multi-Modal Modeling, Analysis and Validation of Open Source

Software Development Processes. *Intern. J. Internet Technology and Web Engineering.* 1(3), p. 49-63, 2006.

[85] Scacchi, W. Free/Open Source Software Development: Recent Research Results and Empirical Methods. In *Advances in Computers*, Zelkowitz, M.V. ed. 69, p. 243-295, 2007.

[86] Simone, C., Mark, G., et al. Interoperability as a Means of Articulation Work. In *Proceedings of the ACM Conference on Work Activities Coordination and Collaboration (WACC '99).* p. 39-48, ACM Press. San Francisco, CA, February 22-25, 1999.

[87] Singhal, S. and Zyda, M. Networked Virtual Environments: Design and Implementation. Addison-Wesley: New York, NY, 1999.

[88] Star, S.L. and Griesemer, J.R. Institutional Ecology, 'Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39. *Social Studies of Science.* 19, p. 387-420, 1989.

[89] Star, S.L. and Ruhleder, K. Steps Toward an Ecology of Infrastructure: Design and Access for Large Information Spaces. *Information Systems Research.* 7(1), p. 111-134, 1996.

[90] Stiemerling, O. and Pacull, F. Secure Component-Based Groupware. In *Proceedings of the CSCW 2000 Workshop on Component-Based Groupware.* p. 43-48, Philadelphia, PA, December, 2000.

[91] Suryanarayana, G., Erenkrantz, J.R., et al. PACE: An Architectural Style for Trust Management in Decentralized Applications. In *Proceedings of the 4th Working IEEE/IFIP Conference on Software Architecture.* p. 221-230, Oslo, Norway, June, 2004.

[92] Suryanarayana, G. and Taylor, R.N. A Survey of Trust Management and Resource Discovery Technologies in Peer-to-Peer Applications. Institute for Software Research, University of California, Irvine, Technical Report UCI-ISR-04-6, July, 2004.

[93] Suryanarayana, G., Erenkrantz, J.R., et al. An Architectural Approach for Decentralized Trust Management. *IEEE Internet Computing.* 9(6), p. 16-23, November/December, 2005.

[94] Suryanarayana, G., Diallo, M.H., et al. Architectural Support for Trust Models in Decentralized Applications. In *Proceedings of the 28th International Conference on Software Engineering (ICSE 2006).* p. 52-61, Shanghai, China, May, 2006.

[95] Suryanarayana, G. and Taylor, R.N. TREF: A Threat-centric Comparison Framework for Decentralized Reputation Models. Institute for Software Research, University of California, Irvine, Technical Report UCI-ISR-06-2, January, 2006.

[96] Suryanarayana, G. and Taylor, R.N. SIFT: A Simulation Framework for Analyzing Decentralized Reputation-based Trust Models. Institute for Software Research, University of California, Irvine, Technical Report UCI-ISR-07-5, August, 2007.

[97] Van House, N.A., Butler, M.H., et al. Cooperative Knowledge Work and Practices of Trust: Sharing Environmental Planning Data Sets. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW '98).* p. 335-343, ACM Press, New York, NY. Seattle, WA, November 14-18, 1998.

[98] Vince, J. and Earnshaw, R. Virtual Worlds on the Internet. IEEE Computer Society, IEEE Press: Los Alamitos, CA, 1998.

[99] Walsh, K. and Sirer, E.G. Thwarting P2P Pollution Using Object Reputation. Department of Computer Science, Cornell University, Report cul.cis/TR2005-1980, 2005.

[100] Wenger, E. Communities of Practice. Cambridge University Press, 1998.

[101] Yao, W.T.-M. Fidelis: A Policy-Driven Trust Management Framework. In *Proceedings of the 1st International Conference on Trust Management.* 2692, p. 301-317, Springer-Verlag Heidelberg. Heraklion, Crete, Greece, May 28-30, 2003, 2003.

[102] Yin, R.K. The Case Study as a Serious Research Strategy. *Knowledge: Creation, Diffusion, Utilization.* 3, p. 97-114, 1981.

[103] Yin, R.K. Case Study Research, Design and Methods. 2nd ed. ed. Sage Publications: Newbury Park, CA, 1994.

[104] Yu, B. and Singh, M.P. A Social Mechanism of Reputation Management in Electronic Communities. In *Proceedings of the Fourth International Workshop on Cooperative Information Agents.*

p. 154-165, 2000.

[105] Zacharia, G. and Maes, P. Collaborative Reputation Mechanisms in Electronic Marketplaces. In *Proceedings of the 32nd Hawaii International Conference on System Sciences.* Hawaii, 1999.

[106] Zacharia, G. and Maes, P. Trust Management Through Reputation Mechanisms. *Applied Artificial Intelligence.* 14, p. 881-907, 2000.

[107] Zimmerman, A. and Nardi, B. Whither or Whether HCI: Requirements Analysis for Multi-Sited, Multi-User Cyberinfrastructures. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI'06).* p. 1601-1606, Montreal, Ontario, Canada, 2006.

[108] Zuboff, S. In the Age of the Smart Machine. Basic Books: New York, 1988.

[109] Julsrud, T. Core/periphery Structures and Trust in Distributed Work Groups: A comparative case study. *Structure and Dynamics: eJournal of Anthropological Related Sciences.* Forthcoming.

Game-Based Virtual Worlds as Decentralized Virtual Activity Systems

Walt Scacchi

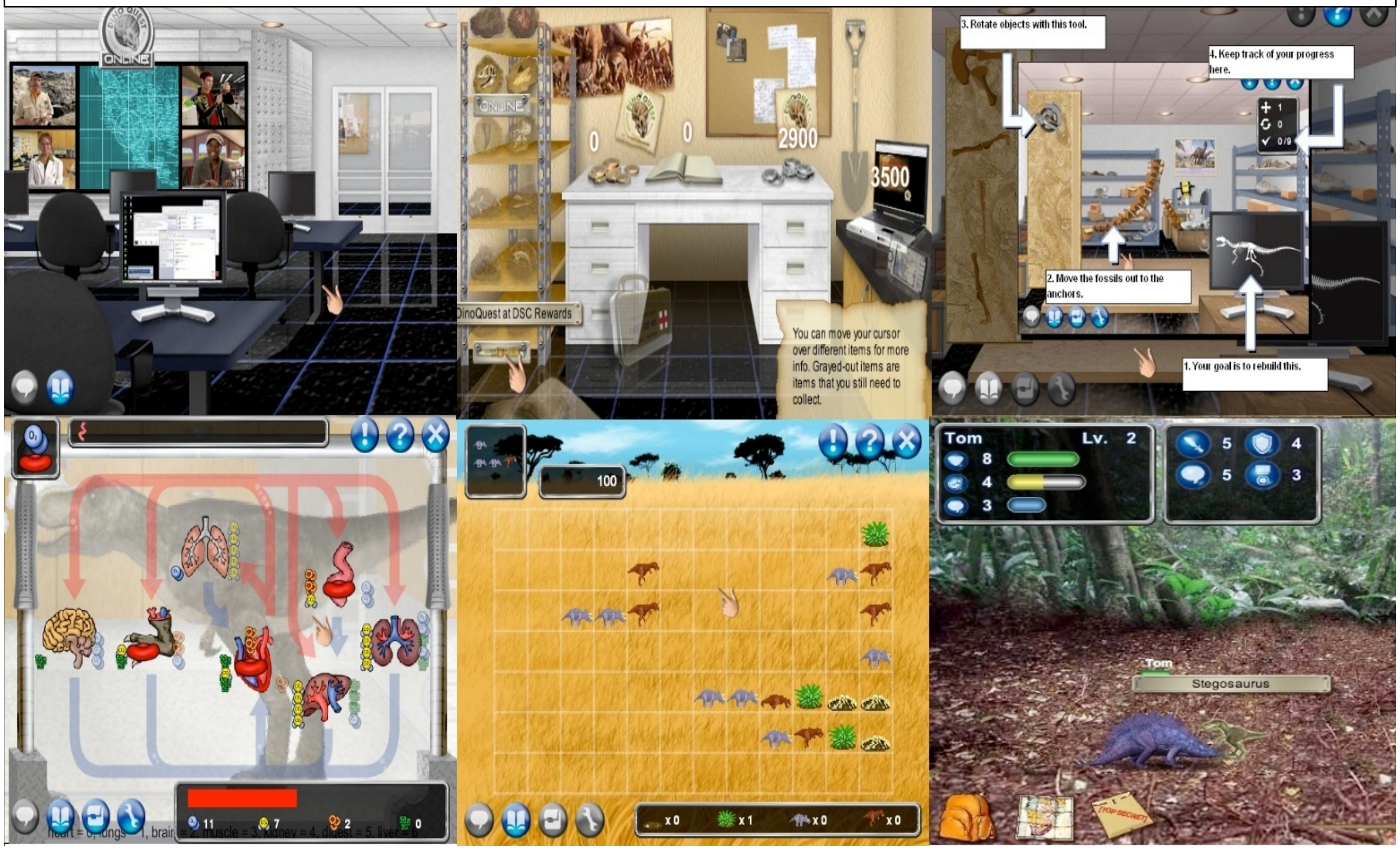
There is widespread interest in the development and use of decentralized systems and virtual world environments as possible new places for engaging in collaborative work activities. Similarly, there is widespread interest in stimulating new technological innovations that enable people to come together through social networking, file/media sharing, and networked multi-player computer game play. A *decentralized virtual activity system* (DVAS) is a networked computer supported work/play system whose elements and social activities can be both virtual and decentralized (Scacchi et al. 2008b). Massively multi-player online games (MMOGs) like *World of Warcraft* and online virtual worlds like *Second Life* are each popular examples of a DVAS. Furthermore, these systems are beginning to be used for research, development, and education activities in different science, technology, and engineering domains (Bainbridge 2007, Bohannon et al. 2009; Rieber 2005; Scacchi and Adams 2007; Schaffer 2006), which is also of interest here. This chapter explores two case studies of DVAS's developed at the University of California at Irvine that employ game-based virtual worlds to support collaborative work/play activities in different settings. The settings include those that model and simulate practical or imaginative physical worlds in different domains of science, technology or engineering through alternative virtual worlds where players/workers engage in different kinds of quests or quest-like workflows (Jakobsson 2006).

Each of the two case studies is presented in a manner that identifies a number of themes or variables that are used for comparative analysis. This analysis seeks to identify relationships between how development and usage variables are intertwined, to understand how development shapes subsequent usage, and how anticipated usage shaped development. Said differently, DVAS's are socio-technical systems, so to understand and compare their development and use helps draw attention to the socio-technical interaction networks and processes that emerge along the way (Scacchi et al. 2008b). The variables of interest include: the target science, technology, or engineering domain; representative activities performed within the domain through games or virtual worlds; how they are used to support learning; what kinds of social, technological, or educational affordances are employed to facilitate collaborative activities (Kirschner et al. 2004; Scacchi et al. 2008a); and integrated or situated experiences (rather than disjoint system functions, computational services, or system capabilities) that arose through these activities. Finally, there is a discussion of outcomes and surprises that emerged from the development and use of these systems in their respective contexts of use.

Case 1: Science Learning Games for Informal Life Science Education

The first case is from a game-based virtual world called *DinoQuest Online* (DQO). DQO was designed for informal science education in the domains of life science and paleontology for K-6th grade students (Scacchi et al. 2008a). DQO is a free-to-play, science learning game environment deployed on the Web at http://www.DQOnline.org. It was implemented using Flash, and the environment runs within common Web browsers on modest power (or older) personal computers connected to the Internet. Example screenshots from DQO appear in Figure 1, from left to right and top to bottom: (a) DQO virtual collaboratory; (b) player's research results collection space; c) in-game tutorial for how to use game controls during skeletal re-assembly tasks; next row (d) screen of game for exchanging oxygen, CO_2, and nutrients through the cardio-pulmonary system; (e) Tetris-like prey-predator game; (e) DinoSphere multi-player environment that simulates multiple creatures in different ecological niches (Scacchi et al. 2008a).

Figure 1: DinoQuest Online



DQO was created to complement and interoperate with a mixed reality, game-based science exhibit called *DinoQuest*. We also participated in its design, addressing similar issues at the Discovery Science Center (DSC) in Santa Ana, CA.[1] Critical to the design of this game world was its focus on embodying Californian and national science education standards (NSES 1996) for the life sciences in grades K-6. During design activities, our focus was to create what we call *science learning games* that are both fun and scientifically grounded, rather than providing simply an entertaining but inauthentic or misleading characterization of scientific concepts and work practices (cf. Bohannon et al. 2009).

Life science is a foundational area of education for young students, as it helps provide evidence-based approaches for understanding and reasoning about the development, survival, and evolution of living beings. This in turn serves as a basis for understanding human health and reasoning about living systems in the world around us, among other things (NSES 1996). However, there are many challenges for how best to present such concepts in ways that are readily accessible to students in age, skill-level, and school grade appropriate manners. Though the study of dinosaurs is only a small part of the study of pre-historic life (paleontology), children are widely found to exhibit interest and curiosity in dinosaurs, and so our choice was to develop science learning games for life sciences for young learners that employ dinosaurs as characters whose in-game activities are mediated or expressed through their life systems and processes. These systems and processes are designed as analogs of those found in humans or living creatures. Thus, activities central to successful play of DQO entails a variety of identification, recognition, discovery, interactive manipulation, and reasoning tasks that are scaffolded through in-game human characters that serve as collaborators and role models. These characters serve in different roles as scientists, specialists, or technicians who provide prompts, cues, and feedback (acknowledgement of accomplishments, or suggested alternative actions to take in response to failures) to players. A host of other features support informal science education including topical graphics, animated visualizations, music and audio cues, situated tutorials and in-game help, multi-genre games and game play mechanics, progress and resource utilization scores (via in-game dashboards), and collaboration affordances (Clark and Mayer 2008; Rieber 2004; Scacchi et al. 2008a).

Figure 1 provides some examples of these through a collage of in-game screenshots. Starting from Figure 1a, play begins on entry into an in-game world that visually suggests a setting where computing and telecommunications activities occur, including a tiled, multi-screen display display with different in-game human characters (scientists) can be engaged, who each need assistance in solving problems at hand. These problems are embodied as mini-games, and a total of 13 are included, for about 3-5 hours of total game play. Next, to the right in Figure 1b, each player has their own research space where their research (game play) results will appear, so they can keep track of their progress and goals obtained or to be obtained in order to advance to more challenging games (i.e., multi-level game play). The remaining screenshots in Figure 1 highlight other increasingly challenging games whose completion requires accumulative mastery of fundamental life science concepts.

There are a variety of science learning experiences encountered while playing through DQO. Game play is partially ordered and leveled, so that early experiences establish the foundations of play and scientific concepts that need to be employed at later stages and higher challenge levels. Games include digging up dinosaur skeleton fossils whose configuration and orientation are hidden, identifying and classifying different skeletal bones or substructures, reconfiguring and assembling skeletal components into recognizable creature forms, as well as others that exhibit concepts for how balance, proportion, and size affect the speed with which a creature travels, prey-predator relationships, and more. In each game, players act in the role of research assistants who help visually and aurally depicted scientists or technicians from different disciplines to collect data, compose

1 http://www.discoverycube.org/exhibit.aspx?q=11

artifacts, observe relationships, and experience decision-making or problem-solving trade-offs. When the student is unable to advance during game play due to errors, mistakes, or gaps in understanding, the in-game science characters offer guidance or reasoning tips to help scaffold the player toward discovering or engaging the causal relationships that provide the path forward. In-game textual help and tutorials are also provided, though their usage seems primarily of value to adults (parents or teachers) who do not understand how the game works, or who want to collaborate with, or work over the shoulder of their children in enacting science learning through game play (Twidale 2005). Finally, play in the final level DinoSphere mini-game (displayed in Figure 1f) entails directing the in-game activities of simulated dinosaurs who react to the situation and surrounding environments (visually depicted) they are in, which generally includes goals like finding food and surviving (in the presence of limited food and/or predators), which can include collaborating with other players' dinosaur characters to fulfill such goals. Game play here is modeled after *The Sims*, in that players direct their in-world characters (dinosaurs like a stegosaurus or velociraptor) in different ecological niches where different kinds of life sustaining behaviors may be important (e.g., finding food, eluding predators, socializing with other dinosaurs like theirs for group activities including hunting for food/prey or overwhelming predators).

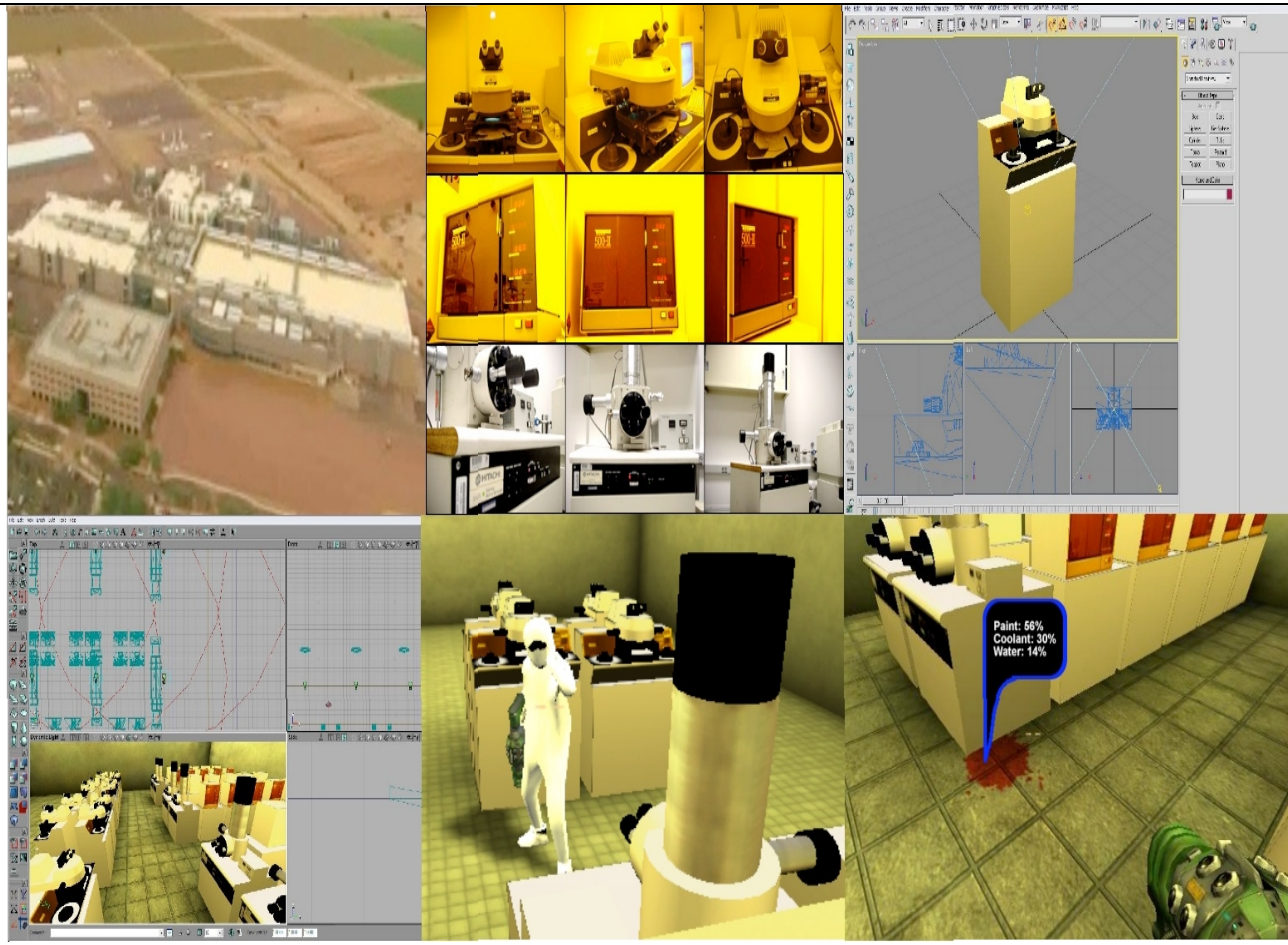
Finally, as suggested in the upper left corner of Figure 1, the DQO virtual world is situated within a virtual workplace that incorporates a wall-sized, multi-tile display that young players navigate in ways similar to an online scientific research collaboratory (Olson et al. 2008; Scacchi et al. 2008a). Young people, after all, need to learn about modern scientific work practices and instrumentalities if we hope for them to develop an interest in a possible career in science when they get older and consider college-level education. So both DQO and the DinoQuest interactive exhibit at DSC are designed to embody and reflect how scientists in field sites might collaborate over multi-media communication networks with colleagues in other disciplines in the course of their work practices (cf. Olson et al. 2008). Further information how this system supports various kinds of collaborative science learning activities and affordances can be found elsewhere (Scacchi et al. 2008a).

Case 2: Game Mod for Semiconductor Fabrication Operations and Service Training

The next case employs a custom-built game "mod" that creates a virtual world for the domain of semiconductor (or nanotechnology) manufacturing using a retail computer game, *Unreal Tournament* (Brown and Scacchi 2007). The resulting game, called *FabLab*, is highlighted in Figure 2, and a demo video is available (Brown and Scacchi 2008). The illustration shows (a) an aerial view of Intel's Fab32 factory; (b) photographs of advanced manufacturing devices; (c) a CAD model of a manufacturing device; (d) a UT software development kit (SDK) for configuring the virtual manufacturing laboratory; (e) inside the FabLab virtual laboratory world, with trainee avatar; (f) a scene from the FabLab spill diagnostic scenario.

We took the standard UT game and content assets and modified them to model, visualize, and simulate the workplace and work activities of technicians who operate and service complex manufacturing systems found in costly cleanroom factories (Brown and Scacchi 2007; Intel Education 2009). In contrast to DQO, focus here is directed developing and deploying a game-based learning environment targeted to adults recently hired to become fabrication technicians, and to provide such training in a way that can scale to multiple, globally dispersed locations and workforces.

Figure 2: FabLab Game



More than one hundred large-scale semiconductor fabrication factories are in operation worldwide. Many now cost more than $1,000,000,000 just to design and build, and companies like Intel operate dozens. Training technicians who can work competently with different manufacturing machines and processes can take years of elapsed time, and many such factories require thousands of technicians who work in shifts distributed 24x7x365 in order to satisfy global marketplace demands for innovative semiconductor devices. While on-the-job training is widespread, there is ongoing need to develop new training materials and experiences that can both streamline the time and cost of deploying such training, and thus there is great interest in e-learning systems and capabilities (Clark and Mayer 2008; Schank 2001).

UT is a game designed as an action-oriented, "first-person shooter" (FPS) style, multi-player game world. It provides ready-to-use functionality for up to 32 concurrent players, who can play over a network/Internet, with built-in support for in-game text or voice chat. It also includes an end-user extensible game engine (a programmable client-server run-time environment), and a game SDK. Using the SDK, it is possible to modify the existing game levels, game play rules/action scripts, and other contents/assets, and all such mods can be redistributed as free/open source software (Scacchi 2004). However, a licensed copy of UT is needed to play a modded game like *FabLab*. Working with a modded version of UT, we could create a game-based virtual world for semiconductor manufacturing, where game play is organized around fabrication technician training activities, operational and service interactions with manufacturing devices, master technician to trainee technician interactions, collaborative multi-technician diagnostic activities, and other workplace scenarios.

Work in an advanced manufacturing facility like a semiconductor fabrication laboratory entails many kinds of training and operational activities. One category of foundational training activities centers on new technicians learning how to prepare themselves for entry into a cleanroom environment, so as not to introduce contaminants that might compromise the integrity of microscopic or nano-scale manufacturing processes or equipment. As revealed in existing traditional text-based training materials, putting on a cleanroom gown (or "bunny suit") entails dozens of steps in specific locations, some in certain postures or body positions (Intel Education 2009). Modeling and simulating a cleanroom gowning process requires the creation of assets, in-game character behavior/animation scripts, and more as mods, as none of these features are part of the UT game. However, the FabLab game support the ability to study, walk-through, or rehearse the gowning process. This helps avoid potentially awkward learning experiences associated with getting (un)dressed in a new workplace with new or unfamiliar co-workers, as well as minimizing the cost of manufacturing problems that emerge from the introduction of contaminants that may unintentionally be brought in by technicians.

Another particularly vexing and challenging problem for such settings is how to collaboratively diagnose breakdowns in operations or complex equipment in geographically remote locations. This game mod demonstrates how manufacturing breakdowns due to faulty equipment or unanticipated materials spills/leaks can be modeled and collaboratively diagnosed, either locally and at a distance, as well as be used in training new manufacturing technicians (Brown and Scacchi 2007, 2008). For example, Figure 2f shows a training scenario where a factory technician locates a liquid spill near a scanning electron microscope, and must determine whether the spill is associated with this device, and whether it involves a hazardous material. The trainee player can use a remote sensing instrument that provides a focus reticule that is aimed at the spill, and the trainee interprets the visual evidence and instrument readings to develop a diagnosis. If the problem lies with the device under scrutiny, the trainee has the potential to call up an animation that depicts the disassembly of the device for servicing. In contrast, in a remote diagnosis, master diagnostic technicians in one location can assist technicians on site in a remote factory location through networked game play mechanisms, when the remote factory equipment layout is configured to reflect current operations at the remote site. In both cases, players can use voice chat and mobile PCs to collaboratively engage in diagnosing visually observed evidence to determine possible causes and appropriate remedies/interventions.

Discussion

Many pleasant surprises arose during the development of DQO while working with our sponsors and collaborators at the DSC. Unpleasant surprises, grounded in dilemmas common to game and game software development (e.g., schedule overruns or poorly documented software functions), are left out of the discussion here.

First, designing games that address explicit education standards – like the NSES (1996) – turned out to be quite liberating from a game design perspective. These standards helped make clear to the DQO developers what learning goals were needed and appropriate for learners of different ages/skill levels. The standards also highlight dependencies among concepts, which we found helped to simplify the challenges of what game play mechanics or game genre to employ to convey, embody, or experience specific science concepts. An example here is our repurposing of a Tetris game and play mechanics to depict prey-predator and food cycle relations (as displayed in Figure 2e), and awards in-game points to players who correctly match these relations as new dinosaurs enter the play space for sorting and matching.

Second, during early usage evaluation and feedback studies at DSC, we found we needed to support parents and teachers who experienced difficulty in comprehending what was going on (e.g., what scientific concepts or relations are in focus, how game play works, how in-game controls operate, how points and resource utilization are scored), while young players would readily dive into game play and start solving game play problems. Such support was subsequently developed and integrated (see Figure 1c), and this helped facilitate better parent/teacher-child collaborative learning, based on our observations and user feedback.

Third, we found that some young students are able to provide copious explanations about what is going on and how the game works to their adult companions, while others provide much less. Though we collected many examples of these during evaluations of DQO game play at DSC, it seems that science learning games like DQO can become more effective learning environments when they provide in-game mechanisms that elicit or encourage online discourse and questions that elicit age/skill appropriate written explanations to further improve and deepen the value of the scientific concepts that have been learned. Such accomplishments have recently been demonstrated in other science learning game environments (cf. Shaffer 2006; Schaller et al. 2009).

In contrast to DQO, many technical and research challenges emerged with the development of the FabLab game world. First, when the project began, our focus was to respond to a challenge from our industrial project collaborators and sponsor (Intel). The challenge was to identify potential refinements and applications for how best to support globally distributed project teams who could interact and manipulate shared online artifacts and tools through domain-independent, collaborative virtual worlds (cf. Pickering et al. 2006) or other tools for visualizing socio-technical interaction networks whose members/elements were decentralized. This focus eventually led to a group of analysts and training personnel involved in finding ways for how to scale-up and optimize the training of thousands of technicians that are needed to operate and service a new semiconductor fabrication facility. Prior experience with multi-player FPS games brought to mind numerous in-game worlds (or levels) that situated game play in virtual laboratories, factories, or underground industrial infrastructures. Recognizing this, along with the ways and means for modding such game-based virtual worlds, quickly pointed the way for what could be modded to recreate a domain-specific, modern semiconductor fabrication facility in which multi-player activity could produce complex work practices and situations that can be (re)mediated through player-directed in-game avatars. So, we quickly moved to mod a capable FPS game (UT) to effectively create a low-cost, game-based work practices simulator for semiconductor fabrication service training that could be readily deployed in a multi-player environment that could operate over the Internet or local-area network.

Second, our experience with the underlying game and game play mechanics of UT also gave rise to discovering new ways to collaboratively diagnose operational problems like material spills or contamination across remote, networked facilities. Specifically, event-driven game play mechanics are often used to affect activities within the game world like opening a door when to touch (i.e., proximity detection) its handle, or enabling an interaction with a non-player character or in-world object whose behavior (e.g., an explosion that distributes object elements within a limited range) mediates what a player's avatar can do next. Similarly, by repurposing in-game FPS weapons to sense instead of shoot, allows a conceptual overloading of familiar play objects (weapons) to serve more instrumental and constructive ends. Using these, it then became possible to develop training scenarios where a taxonomy of detrimental or potentially hazardous material spills could be articulated, a given spill type could introduced essentially anywhere on/near a modeled manufacturing machine, and players could sense and diagnose the problem and determine an appropriate response (e.g., service a leaky manufacturing machine) or voice chat with another collaborator (technician trainer or remote consulting technician). Such capabilities represent a new technological innovation in semiconductor fabrication training and operational service, and such an innovation ultimately emerged from modification and repurposing of games, game play, and game play mechanics.

Conclusions

Overall, this chapter seeks to articulate and explore how game-based virtual worlds can enable new modes of collaborative experience in science, technology, and engineering domains where decentralized play-as-work and work-as-play activities can emerge. The domains of informal life science education for young learners acting in regional science centers or at home, and for training adult technicians in the operation and service of advanced, high technology manufacturing systems are each of practical import and high consequence. Each further demonstrates the range and diversity of activities and collaboration affordances within virtual worlds, as well as across domains for scientific research and education, that can be enhanced through collaborative play-work (Bainbridge 2007).

Game-based virtual worlds can be employed in ways that support scientific research practices, technological innovation, and development of advanced engineering/manufacturing systems. Science and technology oriented game-based virtual worlds like *DinoQuest Online* and *FabLab* represent an interesting experiment in the collaborative construction (O'Donnell 2009) and use of decentralized virtual activity systems that different audiences find provide playful, productive, and collaborative interactions and learning/discovery-focused quests.

Acknowledgements

Support for the research is supported through grants from the National Science Foundation #0534771 and #0808783; Digital Industry Promotion Agency, Global Research and Development Collaboration Center, Daegu, South Korea; Discovery Science Center; and Intel Corporation. No endorsement implied. Special thanks to Robert Nideffer, Alex Szeto, and Craig Brown at the UCI GameLab; Joe Adams and Janet Yamaguchi at DSC; as well as Eduardo Gamez and Eleanor Wynn at Intel, for their participation, contributions, and encouragement.

References

Bainbridge, William Sims. 2007. "The Scientific Research Potential of Virtual Worlds." *Science* 317: 472-476.

John Bohannon, T. Ryan Gregory, Niles Eldredge, and William Sims Bainbridge. 2009. "Spore: Assessment of the Science in an Evolution-Oriented Game." In *Online Worlds: Convergence of the Real and Virtual*, edited by W. S. Bainbridge (this volume). London: Springer.

Brown, Craig, and Walt Scacchi. 2007. "Modeling and Navigating a Simulated Semiconductor Fabrication Laboratory using a Game Engine," http://www.ics.uci.edu/~wscacchi/GameLab/FactoryVisualization/FabLab-Draft-Scacchi-15Oct07.pdf

Brown, Craig, and Walt Scacchi. 2008. *FabLab Game Demo Reel#3.* http://www.ics.uci.edu/~wscacchi/GameLab/DemoReels/FabLab-Demo-03.wmv

Clark, Ruth Colvin, and Richard E. Mayer. 2008. *e-Learning and the Science of Instruction: Proven Guidelines for Consumers and Designers of Multimedia Learning*. San Francisco: Pfieffer.

Intel Education. 2009. *What is a Cleanroom? Working in a Cleanroom*. Intel Corporation, http://www.intel.com/education/cleanroom/index2.htm, accessed 15 April 2009.

Jakobsson, Mikael. 2006. "Questing for Knowledge – Virtual Worlds as Dynamic Processes of Social Interaction." Pp. in 209-225 in *Avatars at Work and Play*, edited by R. Schroder and A. S. Axelsson. Dordrecht, Netherland: Springer.

Kirschner, Paul, Jan-Willem Strijbos, Karel Kreijns and Pieter Jelle Beers. 2004. "Designing Electronic Collaborative Learning Environments." *Educational Technology Research and Development* 52(3): 47-66.

NSES. 1996. *National Science Education Standards*. Washington, DC: The National Academies Press; http://www.nap.edu/openbook.php?record_id=4962.

O'Donnell, Casey' 2009. "The Everyday Lives of Video Game Developers: Experimentally Understanding Underlying Systems/Structures" *Transformative Works and Culture* 2; http://journal.transformativeworks.org, accessed 15 April 2009.

Olson, Gary M., Ann Zimmerman and Nathan Bos (eds.). 2008. *Scientific Collaboration on the Internet*. Cambridge, MA: MIT Press.

Pickering, Cindy, John David Miller, Eleanor Wynn and Chuck House. 2006. "3D Global Virtual Teaming Environment." Pp. 126-135 in *Proceedings of the The Fourth International Conference on Creating, Connecting and Collaborating through Computing, (C5'06)*. January 26-27, 2006, Berkeley, CA.

Rieber, Lloyd P. 2005. "Multimedia Learning in Games, Simulations, and Microworlds." Pp. 549-567 in *The Cambridge Handbook of Multimedia Learning*, edited by Richard E. Mayer. Cambridge, UK: Cambridge University Press.

Scacchi, Walt. 2004. "Free/Open Source Software Development Practices in the Computer Game Community." *IEEE Software* 21(1): 56-66,.

Scacchi, Walt, and Joe Adams. 2007. "Recent Developments in Science Learning Games for Informal Science Education," presented at *Games, Learning, and Society: 3.0*, Madison, WI. July.

Scacchi, Walt, Robert Nideffer and Joe Adams. 2008a. "A Collaborative Science Learning Game Environment for Informal Science Education: *DinoQuest Online.*" Pp. 71-82 in *New Frontiers for Entertainment Computing* edited by P. Ciancarini, R. Nakatsu, M. Rauterberg, and M. Roccetti. New York: Springer.

Scacchi, Walt, Alfred Kobsa, Crista Lopes, Gloria Mark, Bonnie Nardi, David Redmiles and Richard N. Taylor. 2008b. *Decentralized Virtual Activities and Technologies: A Socio-Technical Perspective*, Institute for Software Research Research Report, UCI-ISR-08-04, December.

http://www.ics.uci.edu/~wscacchi/GameLab/NSF-DVAS-Proposal.pdf

Shaffer, David Williamson. 2006. *How Computer Games Help Children Learn*. New York: Palgrave Macmillan.

Schaller, David T., Kate Haley Goldman, Grant Spickelmier, Steven Allison-Bunnell and Jessica Koepfler. 2009. “Learning In The Wild: What Wolfquest Taught Developers and Game Players.” In *Museums and the Web 2009: Proceedings. Toronto: Archives & Museum Informatics*, edited by J. Trant and D. Bearman; http://www.archimuse.com/mw2009/papers/schaller/schaller.html, accessed 15 April 2009.

Schank, Roger C. 2001. *Designing World-Class E-Learning: How IBM, GE, Harvard Business School, and Columbia University Are Succeeding at E-Learning*. New York: McGraw-Hill.

Twidale, Michael. 2005. “Over the Shoulder Learning: Supporting Brief Informal Learning.” *Computer Supported Cooperative Work* 14: 505-547.

New game concepts and emerging proposal ideas for 2010 and beyond

Ideas for New Money-Making Games:

Possible New UCI-DIP Projects

Walt Scacchi
Game Culture and Technology Laboratory
University of California, Irvine
April 2009

Overview

- Caveats and limitations
- Review of new game ideas from 2007 UCI-DIP video seminar course
- Review of major money-making games at hand
- New money-making ideas for games and game-like virtual worlds
    - MMOSLLGs, Multi-culture STE games, Trans-cultural media production games,
    - Micro-markets for games, Virtual Irvine Spectrum, Super game production environment, Mobile avatars.

Caveats and Limitations

- There is NO way to guarantee that a new game or game-based virtual world will make money.
- Our effort is to present new ideas that offer innovations that build on existing game trends and technologies
- Our goal is to narrow the alternatives presented here into a set that merit further development and refinement
  - We can then develop follow-up proposal and/or business plan for how to proceed.

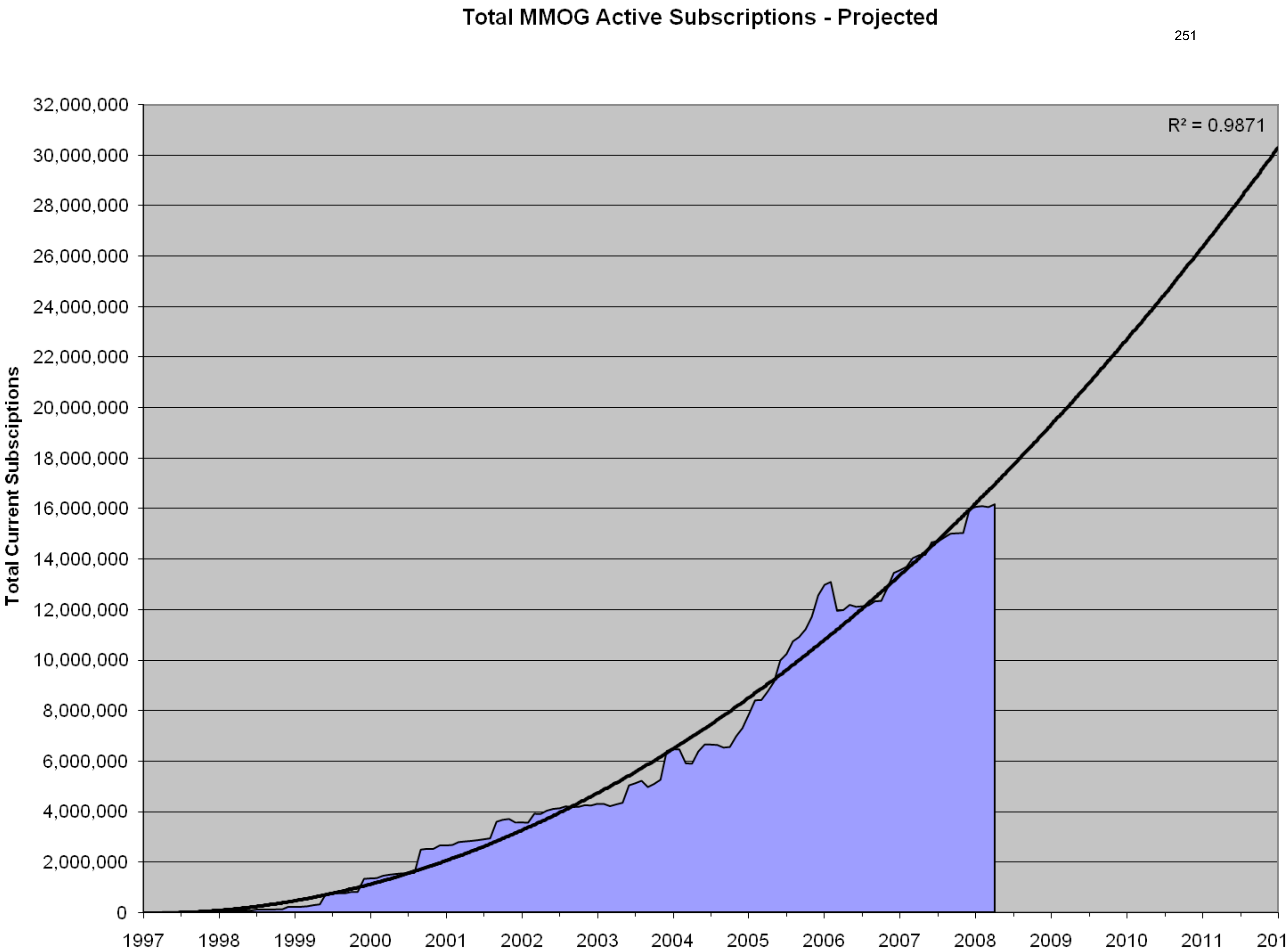
New game ideas from 2007 UCI-DIP video seminar course

- Starting with a review of lecture materials found at: http://www.ics.uci.edu/~wscacchi/GameIndustry/
- New game ideas (c. 2007):
  - MMOG Hybrids (e.g., Casual MMOGs and Game-based Virtual Worlds)
  - Social networking games
  - Combined game types (MMOG and Music, games with heterogeneous play devices)
  - Game Mashups
  - New Game Genres (e.g, location-based games)

Major money-making games: 2009

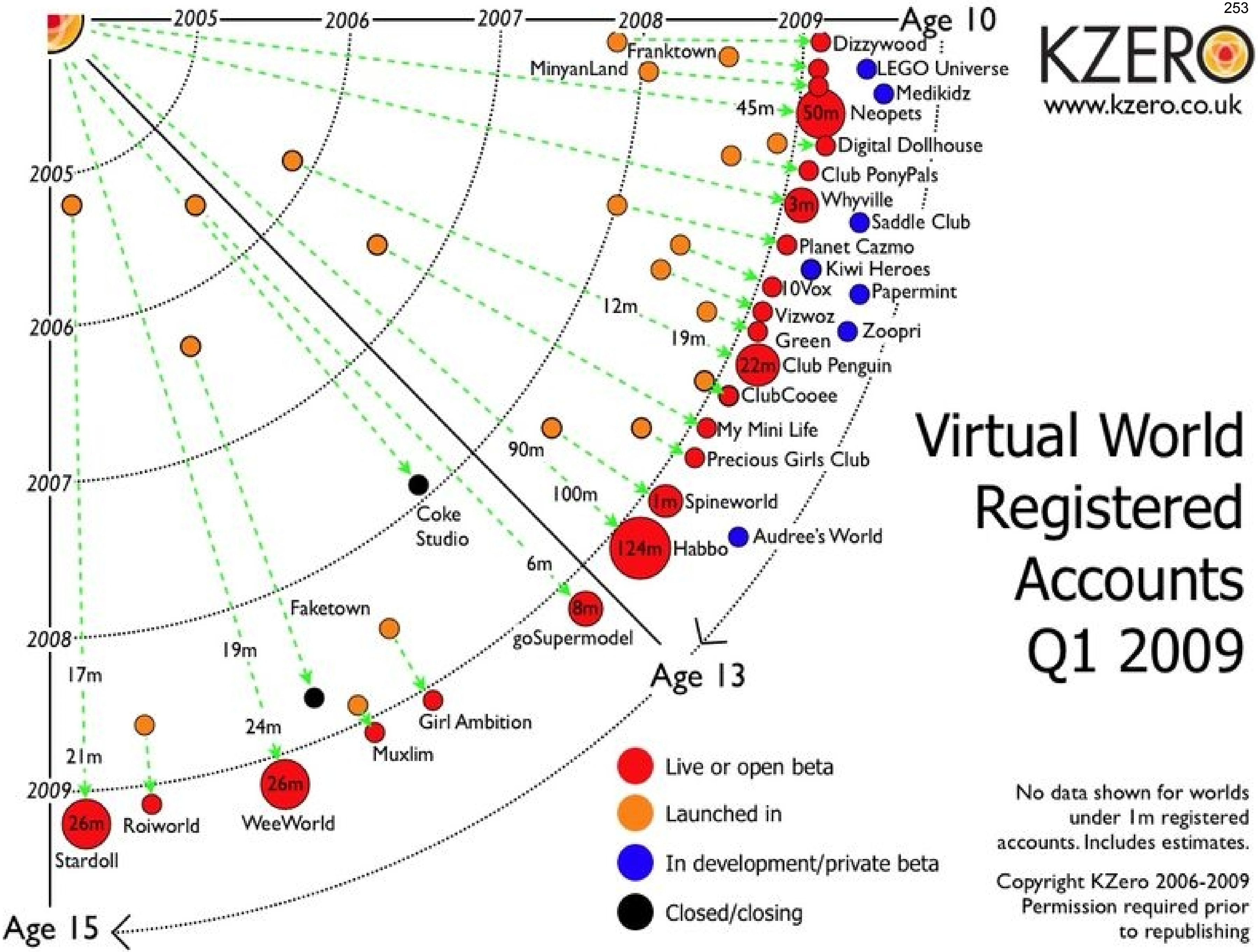
- *World of Warcraft* (11M+monthly subscribers)
- *Grand Theft Auto IV* (13M+ copies sold)
- *The Sims* (100M+ copies sold)
- *Half-Life* (20M+ copies sold)
- *KartRider* (15M+ Korean players)

- *Tabula Rasa* (2007-09) – Major MMOG failure
- *Granado Espada* (Asia) vs. *Sword of the New World* (US+Europe)
  - Asian success, US failure


Total MMOG Active Subscriptions - Projected
R² = 0.9871
Total Current Subsciptions
32,000,000
30,000,000
28,000,000
26,000,000
24,000,000
22,000,000
20,000,000
18,000,000
16,000,000
14,000,000
12,000,000
10,000,000
8,000,000
6,000,000
4,000,000
2,000,000
0
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012

Major money-making games: 2009

- Each arises from a family of related games and game pack extensions
- Each relies on IP created by a single firm, but developed by multiple game studios
- Each relies on high development costs and long-lead development cycles
  - Creates effective barrier to low-cost clones
  - New release launch is critical
- Some being adapted to other media (Cinema)
- Game-based Virtual Worlds may be a larger global market -- http://www.virtualworldsnews.com/research/


2005
2006
2007
2008
2009
Age 10
Age 13
Age 15
KZERO
www.kzero.co.uk
Dizzywood
LEGO Universe
Medikidz
Franktown
MinyanLand
45m
50m
Neopets
Digital Dollhouse
Club PonyPals
3m
Whyville
Saddle Club
Planet Cazmo
Kiwi Heroes
10Vox
Papermint
12m
Vizwoz
Green
Zoopri
19m
22m
Club Penguin
ClubCooee
My Mini Life
90m
Precious Girls Club
100m
1m
Spineworld
Audree's World
124m
Habbo
Coke Studio
6m
8m
goSupermodel
Faketown
19m
17m
Girl Ambition
24m
Muxlim
21m
26m
Roiworld
26m
WeeWorld
Stardoll
Virtual World Registered Accounts Q1 2009
Live or open beta
Launched in
In development/private beta
Closed/closing
No data shown for worlds under 1m registered accounts. Includes estimates.
Copyright KZero 2006-2009 Permission required prior to republishing

New money-making ideas for games and game-like virtual worlds

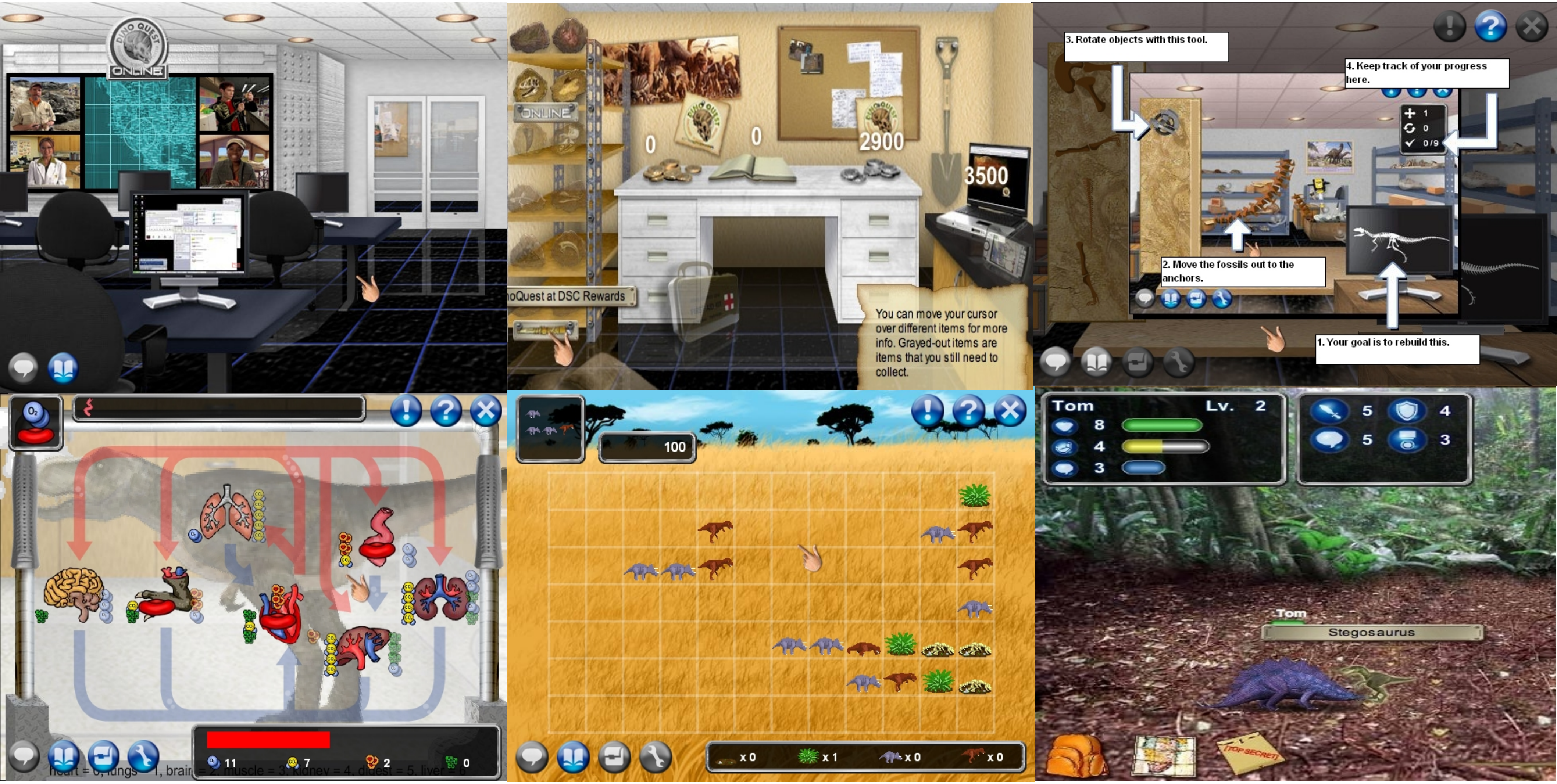
Massively Multiplayer Second Language Learning Games

- **Goal**: Create MMSLLGs (not just another MMOG)
- An MMSLLG is an MMORPG whose play goals is increasing mastery of another natural language (and culture of use)
- Language learning is required to level up
  - Reading, Writing, Speaking
- Requires reading and writing in discussion forums and blogs, around game
  - Peer review and automated language analyses of language skill and appropriate usage
- Supported by indexed videos of game play that provide language construct usage in context.

Multi-Culture Science, Technology, and Engineering Learning Games

- **Goal**: Create a new game market opportunity
  - Global game industry does not develop STE games
- STE games are likely to be attractive to families, educators, and government
- Imagine *DinoQuest Online* – Life Science Game
  - English, Korean, Spanish, Chinese – 250M+ children worldwide as potential player base
- Imagine other STE games for Environment, Energy, Water, Transportation, Advanced Manufacturing, etc.
  - STE game-based virtual world environments?

Web-based science learning games for informal science education for K-6 students and families

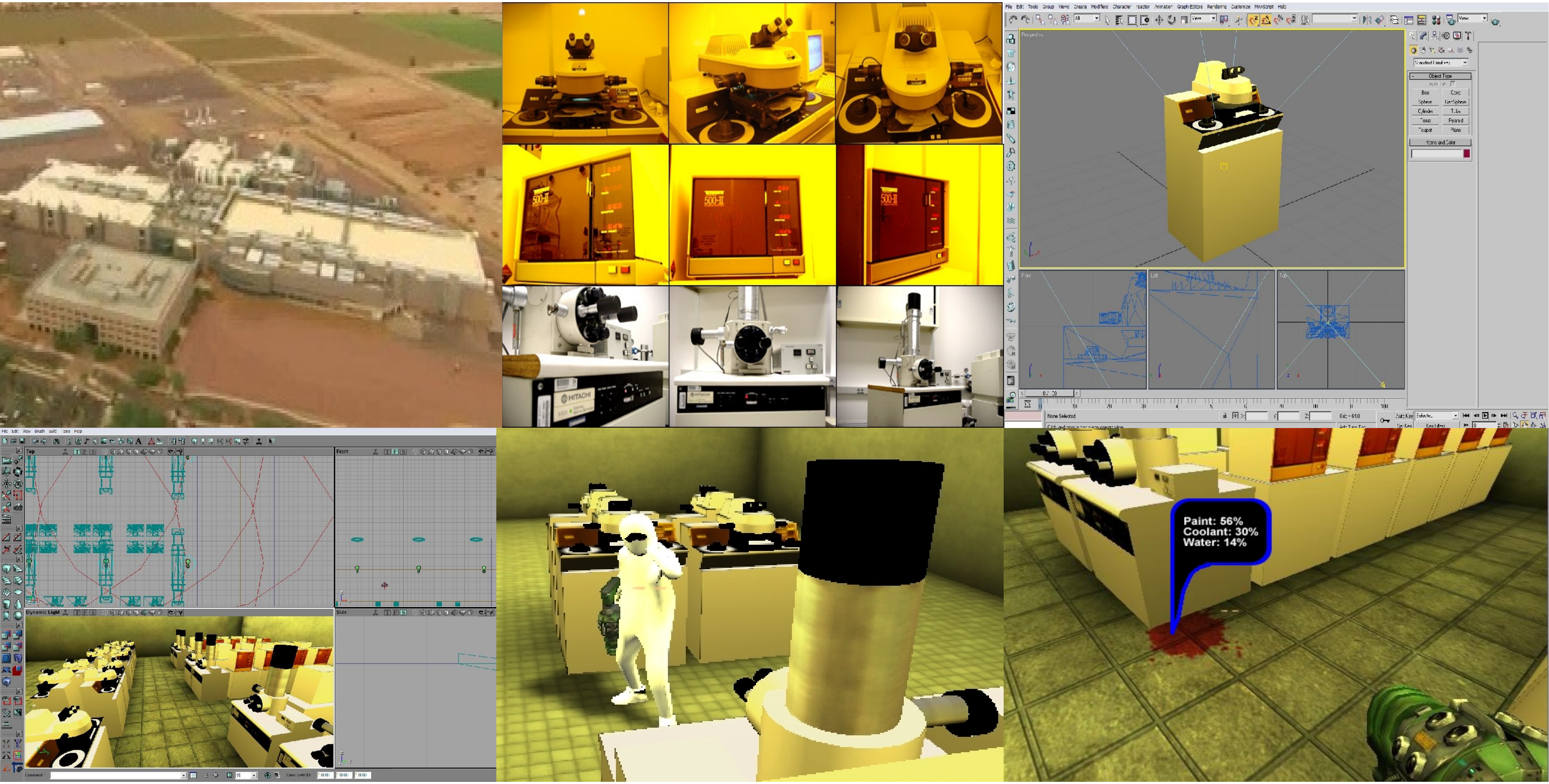


http://www.DQOnline.org/

Multi-Culture Science, Technology, and Engineering Learning Games

- Revenue model based on subsidized deployment and licensing
  - Corporate sponsorship or government
- STE games to be integrated with National Education Program objectives
- STE games to be integrated with National STE Education Standards
  - Facilitates evaluation and adaptive systemic improvement

Semiconductor/nanotech fabrication training game



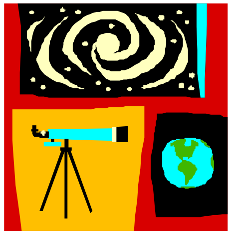
FabLab Demo Reel

Goal: Develop cyberinfrastructure for networked STE learning game and regional STE centers

MMOSTEG Web 3.0 System

Tier 1

Tier 2



Tier 3

Tier 4

Tier 5

Tier 1: Individual player connection: your Internet connection at home.

Tier 2: Local institutional connection: library, science center, school.

Tier 3: Regional science center provides local exhibit content connected online.

Tier 4: “Gateway” science centers provide open interfaces and extensible content.

Tier 5: Science Center Grid: ***Massive Multiplayer Online STE Learning Games and collaboration infrastructure*** for informal K-12 STE education

Remotely Managed Micro-Markets for Games

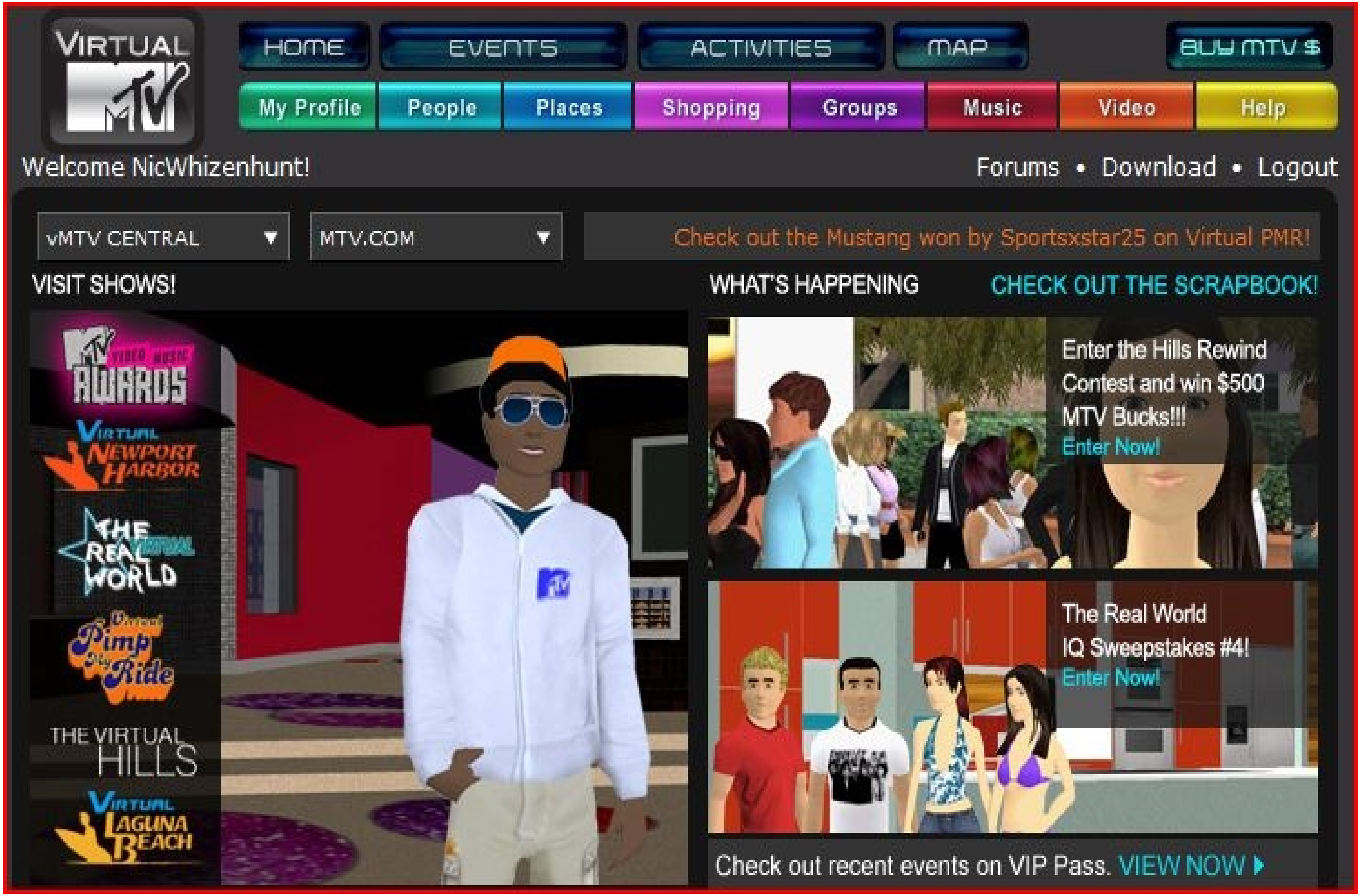
- **Goal**: monetizing game modding or customization
- Create reusable platforms for rapidly creating, deploying, and sustaining micro-markets
  - Micro-market systems can be franchised or licenses to entrepreneurial game modders or operators
- Markets focus on the creation or customization of:
  - Game characters, weapons/objects, levels, play mechanics, rules, and machinima

Virtual Irvine Spectrum

- **Goal**: Create an MMSLLG (seen earlier) application for Asian markets
- Emphasizing immersive *Californian* cultural experience
  - Language, diversity, retail shopping, entertainment and socialization
- Inspired by Virtual Laguna Beach and related media properties from MTV
- Ties to other media (broadcast television programming) and retail merchant product sales
- Separate online merchandising and socialization opportunities for in-world characters


TotallyKatty
BarryWhite
Blaiine
RipTide
lol nice
Check the boxes below to view items on the map.
Place your mouse over the icons for the name of the place
Click on an icon for more info and to visit.
Landmarks
Hot Spots
Shops
Clubs
Where is everyone?
*PREEN

Virtual Irvine Spectrum Concept

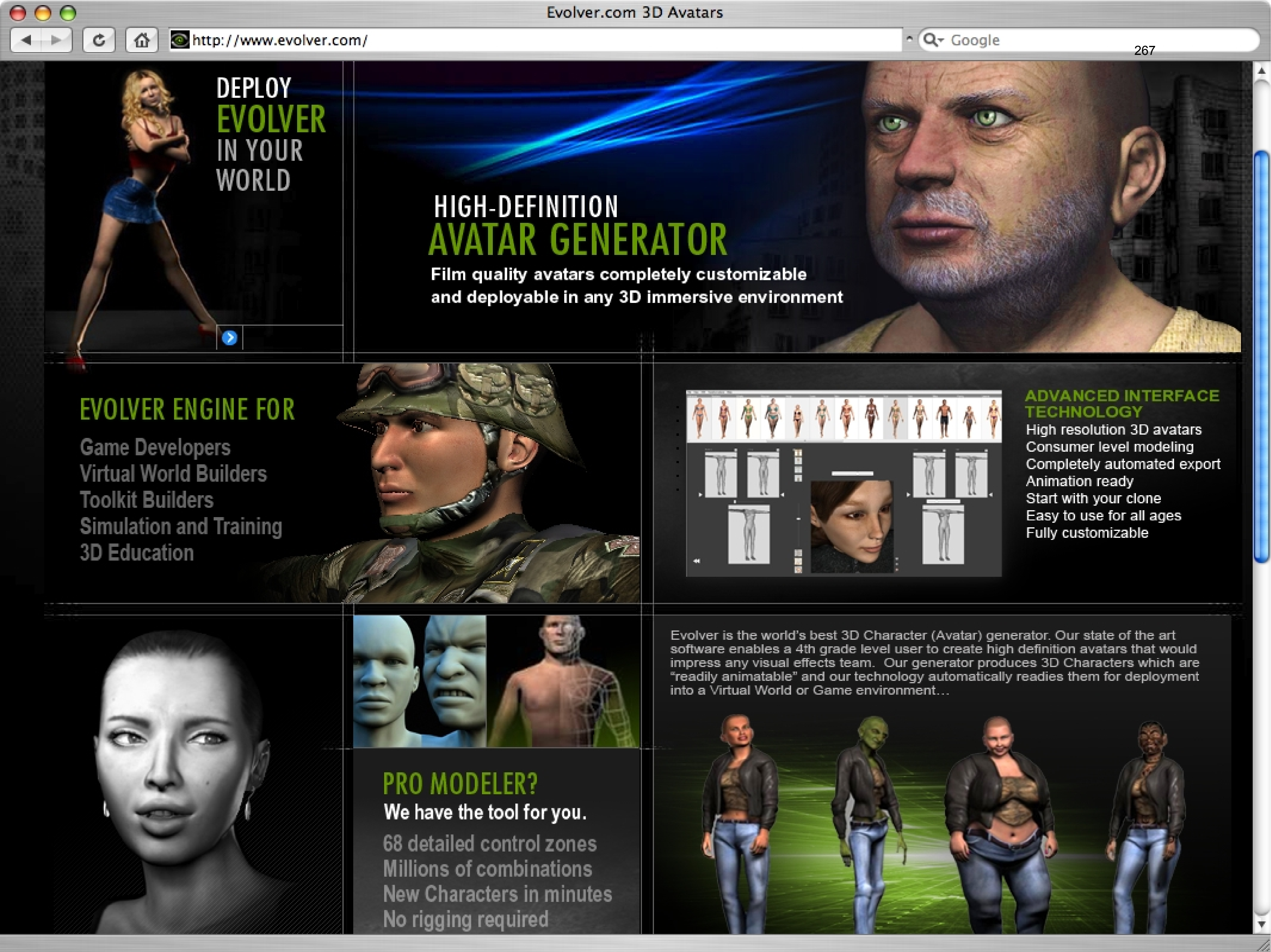


Super game/media production environment

- **Goal**: rapidly produce and populate large-scale games or game-based virtual worlds with emerging IP assets
- Stimulate user created content via micro-markets (seen above)
- Tie-in to other media production worlds
  - *YouTube*, Broadcast television, *Animoto*
- Facilitate large-scale game/VW “ani-jams”
  - A group of developers+artists come together and create one game/VW, by piecing their bounded and interfaced sequences together.
  - May be tied to other mass media (cinema, TV)

Mobile Avatars and Complex Objects

- **Goal**: Game characters or objects that can be moved across different games or VWs
- Requires interlingual (or interworld) game character/object data representation
  - Meta-models for characters/objects and behavioral animation that span targeted universe of interoperating games/VWs
  - Suitable for both user-directed character/object creation and for automated generation
    - Create large populations of diverse characters or objects


Evolver.com 3D Avatars
http://www.evolver.com/
Google
DEPLOY EVOLVER IN YOUR WORLD
HIGH-DEFINITION AVATAR GENERATOR
Film quality avatars completely customizable and deployable in any 3D immersive environment
EVOLVER ENGINE FOR
Game Developers
Virtual World Builders
Toolkit Builders
Simulation and Training
3D Education
ADVANCED INTERFACE TECHNOLOGY
High resolution 3D avatars
Consumer level modeling
Completely automated export
Animation ready
Start with your clone
Easy to use for all ages
Fully customizable
PRO MODELER?
We have the tool for you.
68 detailed control zones
Millions of combinations
New Characters in minutes
No rigging required
Evolver is the world's best 3D Character (Avatar) generator. Our state of the art software enables a 4th grade level user to create high definition avatars that would impress any visual effects team. Our generator produces 3D Characters which are "readily animatable" and our technology automatically readies them for deployment into a Virtual World or Game environment…