Ravages of Time: Synchronized Multimedia for Internet-Wide Process-Centered Software Engineering Environments

Gail Kaiser and Giuseppe Valetto*, Columbia University

{Kaiser, Valetto}@cs.columbia.edu

The emergence of Internet-based software engineering projects and of multimedia software development artifacts introduces new opportunities and challenges for team software development environments, particularly process-centered environments. It is becoming increasingly inexpensive to make audio/video recordings of requirements elicitation meetings with customers, architectural design meetings among co-located development staff, videoconferences among dispersed development staff, etc., and store them on-line for future perusal during incremental development and evolutionary maintenance For practical retrieval, multimedia streams must be segmented and indexed according to semantically meaningful or useful units. However, even assuming excellent search precision and recall, and/or rich hypermedia cross-referencing with conventional development artifacts and process milestone records, there is still the problem of transferring or streaming the multimedia information over the relatively low bandwidth and unassured quality of service (QoS) commodity Internet. Thus, multimedia must also be semantically summarized and compressed, perhaps to be reconstructed using local resources. These are all currently major research topics among image processing and computer vision researchers (e.g., [CMU99, HSG99, JC99, KY98, MYC95, WC99, ZCC96]).

However, those approaches often beg the question as to where the "semantics" come from. They also do not yet address the additional coordination issues that arise when a geographically dispersed team would like to cooperatively employ multimedia for team work, e.g., "watch" video segments in synchrony, so they are seeing and hearing the same things at the same time, (possibly at different resolutions due to their differing local bandwidth and resolution) during their discussion (e.g., via text chat or a separate desktop audio or video conferencing channel).

Consider the scenario of a dispersed development team working in close collaboration on an "open source" project: members may enjoy any combination from 28.8k modem dialup to DSL to cable from their homes or small business offices, to T1 or T3 lines or better at major corporations. *"Dialup will remain the way most people access the Internet right through to the middle of the next decade, Forrester Research believes; but the company foresees at least 26 million broadband users by 2003 and continued growth, and it expects dialup access to peak in 2001 and decline from there"* [Wit99]. WYSIWIS desktop models, such as VNC [RFW98] or other similar tools, become increasingly ineffective across such varied levels of connectivity, with bandwidth becoming a bottleneck due to dispersion of the team and unpredictability of QoS, and are hence infeasible for supporting synchronized streaming multimedia in the foreseeable future

Some combination of prefetching and caching seems mandatory to attain acceptable resolution for remote users as well as maintain close synchrony among distributed teams with widely varying bandwidths and QoS. We hypothesize that one can achieve "smart" prefetching and caching by exploiting the explicit semantic relationships among software artifacts (here, including relevant multimedia or segments thereof) maintained in or referenced by the data repositories of the software development environment, as well as the implicit semantic relationships among software artifacts gleaned through their temporal proximity and/or input/output dependencies with respect to software development tasks described in the process model, or recorded during the process enactment. But to do so effectively, process modeling formalisms and enactment engines may need to be extended with capabilities to directly address "expensive" (in time as well as other resource utilization terms) data accesses or information resources. Related problems may arise with other non-instantaneous and/or open-ended data/resource access/usage, e.g., successively refined Web searches, whether individual or quasi-synchronized during collaborative information filtering.

The Programming Systems Lab at Columbia University has started working together with researchers in computer vision, distributed system resource management and other fields, to build an experimental system (tentatively called Collaborative Content/Context-Adaptive Networked Distance Learning Environment - **C^{3}ANDLE**) to address the following issues:

Process forecasting: Does the "conventional" concept of process/workflow investigated in the software process and workflow communities provide sufficient information (about what will happen "next" or "soon" in the usual case and in likely predicted/handled exceptional cases) to practically guide multimedia delivery adaptation, via caching/prefetching and resource (network, operating system, and application servers and clients) management? How close can we come to meeting the standards of an omniscient "oracle" that magically knows what the users will want to look at and do, and will prefetch/cache the necessary information?

Feasibility of current formalisms: More concretely, are any of the conventional process modeling notations best, or better or worse, for the above purpose? There is as yet no consensus on process modeling notation or enactment paradigm: task graphs, state charts, Petri nets, rules and "process programming" all have their advocates [IPT99]. Here, considering a generalized abstraction for each of the major notational styles, how close can we come to the "oracle"? Does anticipation of "expensive" information access favor one notational style over another? Should the enactment engine "know" that certain information accesses may not be effectively instantaneous? How should this be described in the model and enacted by the engine? Are notational extensions, or a new paradigm altogether needed, in order to account for video and other remotely accessed multimedia, including segment selection and segment replay, as part of a structured dialogue or task?

Multimedia-aware process enactment: If some reasonably small extension of some conventional process modeling paradigm is indeed adequate, how would the relevant enactment be accomplished most efficiently? Should multimedia adaptation directives be included explicitly in the process model to be issued like any other tasks by the process engine? Is this an implicit functionality implemented via an extension or "plugin" to the process engine? Or should an entirely different kind of engine interpret the extended process model notation "in parallel" with its usual enactment?

Role of multimedia in collaborative software engineering: Many professional software engineering projects are very long-lived, and the maintenance personnel who often perform over 50% of the lifecycle effort could arguably benefit from appropriately concentrated access to the upstream requirements and design rationale hidden in email archives and meeting minutes. It seems plausible that access to semantically indexed and segmented recordings of early meetings, particularly stakeholder videoconferences that occur "anyway", could help. Very long-term empirical studies would be needed for definitive answers; we seek at present only to lay the groundwork for an infrastructure allowing such studies to be conducted.

Architectural impact: How does delivery of multimedia and other "expensive" information impact architectures and components for Internet-scale software development environments? Current frameworks apparently assume that most information sharing beyond the LAN involves only rich text documents, relatively small images (e.g., Gantt and PERT charts, architecture diagrams), and binary executables with potentially large but known and fixed footprints, so do the frameworks need to "know" that they must also handle delivery of network bandwidth and local-resource intensive materials? Do environment builders and process administrators need to think about software engineering activities and processes differently in this context?

Our initial empirical investigations will study student team projects in our current Software Engineering and Operating Systems courses, offered periodically over Columbia Video Network – in which lectures are already videotaped, and coordination of geographically dispersed teams carrying out course project assignments is already perceived by both students and teaching staff as a severe problem.

We intend to embed multimedia delivery within a **CHIME** (Columbia Hypermedia IMmersion Environment) 3D virtual world [DK99b], customized to support distance learning in these courses [DPK99]. CHIME's approach to "software engineering over the Internet" was reported at last year's workshop [DK99a]. We plan to manually extract workflows from audit trails of students performing their team project tasks within CHIME, and analyze how well these match the predefined workflows (an approach already investigated by others, e.g., Balboa [CW99]). We will manually analyze how well, in the ideal best case, either predefined or extracted workflows can guide video segment prefetching, both far-in-advance prefetching in the case where the team instantiates an agenda before the meeting, and momentarily-in-advance prefetching in the case where the team determines and/or revises their agenda in the midst of the meeting.

We plan to then select or develop a process enactment system (possibly but not necessarily involving our own Marvel/Oz/Amber system [KF87, BK94, KBP96] and/or more recent "worklet" mobile agent approach [KSD99]) that closely matches these abstract (prose) workflows, define the workflows in that system's notation, and integrate its enactment engine into the CHIME theme world. We envision a combination of c*ontinual validation* (i.e., run time monitoring and predictive control [Sal99]) over the distributed multimedia provision infrastructure administering content to dispersed and heterogeneous clients, based on the state of the clients' workflow as well as their dynamically varying resources, and of a *workgroup cache* [KVD99], which enables prefetching, caching and pushing of likely useful artifacts (multimedia fragments as well as traditional documents) for development groups and individuals, on the basis of historical recorded workflow enactment patterns and semantic inter-artifact relationships.

This implementation will enable us to experimentally repeat the tasks, now driven by the workflow engine, both with live student groups and with simulation of the previously obtained audit trails. We hope this will aid us in developing an automatic mechanism for generating video prefetching directives from the process model, both a priori and while the process is in progress.

* Giuseppe Valetto is currently on leave from CEFRIEL – Milan, Italy.

References

[BK94] Israel Z. Ben-Shaul and Gail E. Kaiser. A Paradigm for Decentralized Process Modeling and its Realization in the Oz Environment. *16th International Conference on Software Engineering*, May 1994. ftp://ftp.psl.cs.columbia.edu/pub/psl/CUCS-024-93.ps.Z.

[CMU99] *Informedia: Researching Digital Video Libraries at Carnegie Mellon University*. November 1999. http://www.informedia.cs.cmu.edu/html/main.html.

[CW99] Jonathan E. Cook and Alexander L. Wolf. Software Process Validation: Quantitatively Measuring the Correspondence of a Process to a Model Using Event-Based Data. *ACM Transactions on Software Engineering and Methodology*, 8(2):147-176, April 1999. http://www.cs.nmsu.edu/~jcook/papers/vjournal.ps.gz.

[DK99a] Stephen E. Dossick and Gail E. Kaiser. Distributed Software Development with CHIME. *ICSE-99 2nd Workshop on Software Engineering over the Internet*, May 1999. http://www.psl.cs.columbia.edu/papers/CUCS-007-99.html.

[DK99] Stephen E. Dossick and Gail E. Kaiser. CHIME: A Metadata-Based Distributed Software Development Environment. *Joint 7th European Software Engineering Conference and 7th ACM SIGSOFT International Symposium on the Foundations of Software Engineering*, September 1999. ftp://ftp.psl.cs.columbia.edu/pub/psl/CUCS-006-99.pdf.

[DPK99] Stephen E. Dossick, Daniel Port and Gail E. Kaiser. *Embedding Model-Based Architecting in a Collaborative Environment*. Columbia University Department of Computer Science, CUCS-016-99, May 1999. ftp://ftp.psl.cs.columbia.edu/pub/psl/ CUCS-016-99.pdf.

[HSG99] L. He, E. Sanocki, A. Gupta and J. Grudin. Auto-Summarization of Audio-Video Presentations. *ACM Multimedia 99*, November 1999.

[IPT99] *Proceedings of the International Process Technology Workshop*. September 1999. http://www-adele.imag.fr/IPTW/.

[JC99] A. Jaimes and S.-F. Chang. Model Based Image Classification for Content-Based Retrieval. *SPIE Conference on Storage and Retrieval for Image and Video Database*, January 1999.

[KBP96] Gail E. Kaiser, Israel Z. Ben-Shaul, Steven S. Popovich and Stephen E. Dossick. A Metalinguistic Approach to Process Enactment Extensibility. *4th International Conference on the Software Process: Improvement and Practice*, December 1996. ftp://ftp.psl.cs.columbia.edu/pub/psl/CUCS-016-96.ps.gz.

[KF87] Gail E. Kaiser and Peter H. Feiler. An Architecture for Intelligent Assistance in Software Development. *9th International Conference on Software Engineering*, March 1987. ftp://ftp.psl.cs.columbia.edu/pub/psl/icse87.ps.gz.

[KSD99] G. E. Kaiser, A. Stone and S. Dossick. A Mobile Agent Approach to Lightweight Process Workflow. *International Process Technology Workshop*, September 1999. ftp://ftp.psl.cs.columbia.edu/pub/psl/CUCS-021-99.pdf.

[KVD99] G. E. Kaiser, C. Vaill and S. Dossick, A Workgroup Model for Smart Pushing and Pulling, *IEEE 8th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, June 1999. ftp://ftp.psl.cs.columbia.edu/pub/psl/CUCS-012-99.zip.

[KY98] J.R. Kender and B.L. Yeo. Video Scene Segmentation Via Continuous Video Coherence. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 1998.

[MYC95] J. Meng, Y. Juan and S.-F. Chang. Scene Change Detection in a MPEG Compressed Video Sequence. *SPIE Symposium on Electronic Imaging: Science & Technology- Digital Video Compression: Algorithms and Technologies*, SPIE 2419, February 1995.

[RFW98] T. Richardson, Q. Stafford-Fraser, K. R. Wood and A. Hopper, Virtual Network Computing, *IEEE Internet Computing*, 2(1): 33-38, Jan/Feb 1998.

[Sal99] John Salasin. *DASADA Proposer Information Pamphlet.* DARPA BAA00-20, December 1999. http://www.darpa.mil/iso/DASADA/DASADA_PIP.html.

[WC99] H. Wang and S.-F. Chang. FaceTrack- Tracking and Summarization Faces from Compressed Video. *SPIE Photonics East*, *Conference on Multimedia Storage and Archiving Systems*, November 1999.

[Wit99] Sarah L. Roberts-Witt. The Coming DSL-Cable Race. *Internet World*, November 15, 1999. http://www.internetworld.com/print/1999/11/15/infra/19991115-dsl.html.

[ZZC96] D. Zhong, H. Zhang and S.-F. Chang. Clustering Methods for Video Browsing and Annotation. *SPIE Conference on Storage and Retrieval for Image and Video Database*, February 1996.