Carnegie Mellon University
Software Engineering Institute

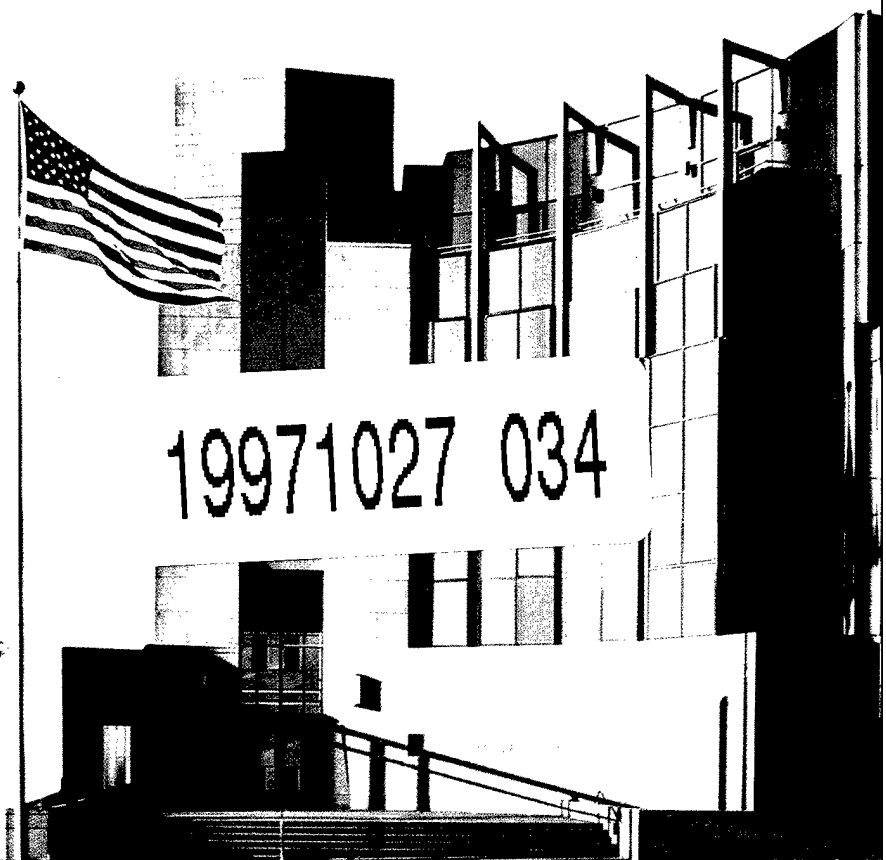
Report of the STEP '97 Workshop on Net-Centric Computing

Scott R. Tilley
Margaret-Anne D. Storey

October 1997

SR

SPECIAL REPORT
CMU/SEI-97-SR-016



Carnegie Mellon University does not discriminate and Carnegie Mellon University is required not to discriminate in admission, employment, or administration of its programs or activities on the basis of race, color, national origin, sex or handicap in violation of Title VI of the Civil Rights Act of 1964, Title IX of the Educational Amendments of 1972 and Section 504 of the Rehabilitation Act of 1973 or other federal, state, or local laws or executive orders.

In addition, Carnegie Mellon University does not discriminate in admission, employment or administration of its programs on the basis of religion, creed, ancestry, belief, age, veteran status, sexual orientation or in violation of federal, state, or local laws or executive orders. However, in the judgment of the Carnegie Mellon Human Relations Commission, the Department of Defense policy of, "Don't ask, don't tell, don't pursue," excludes openly gay, lesbian and bisexual students from receiving ROTC scholarships or serving in the military. Nevertheless, all ROTC classes at Carnegie Mellon University are available to all students.

Inquiries concerning application of these statements should be directed to the Provost, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, telephone (412) 268-6684 or the Vice President for Enrollment, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, telephone (412) 268-2056.

Obtain general information about Carnegie Mellon University by calling (412) 268-2000.

Special Report
CMU/SEI-97-SR-016
October 1997

Report of the STEP '97 Workshop on Net-Centric Computing

Scott R. Tilley
Software Engineering Institute

Margaret-Anne D. Storey
Department of Computer Science, University of Victoria &
School of Computer Science, Simon Fraser University

Reengineering Center
Product Line Systems

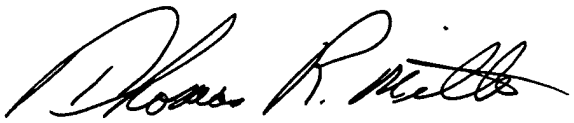
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

This report was prepared for the

SEI Joint Program Office
HQ ESC/AXS
5 Eglin Street
Hanscom AFB, MA 01731-2116

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

FOR THE COMMANDER



Thomas R. Miller, Lt Col, USAF
SEI Joint Program Office

This work is sponsored by the U.S. Department of Defense.

This work was created in the performance of Federal Government Contract Number F19628-95-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 52.227-7013.

This document is available through SAIC/ASSET: 1350 Earl L. Core Road; PO Box 3305; Morgantown, West Virginia 26505 / Phone: (304) 284-9000 / FAX: (304) 284-9001 / World Wide Web: http://www.saic.com/-contact.html / e-mail: webmaster@cpqm.saic.com

Copies of this document are available through the National Technical Information Service (NTIS). For information on ordering, please contact NTIS directly: National Technical Information Service, U.S. Department of Commerce, Springfield, VA 22161. Phone: (703) 487-4600.

This document is also available through the Defense Technical Information Center (DTIC). DTIC provides access to and transfer of scientific and technical information for DoD personnel, DoD contractors and potential contractors, and other U.S. Government agency personnel and their contractors. To obtain a copy, please contact DTIC directly: Defense Technical Information Center / Attn: BRR / 8725 John J. Kingman Road / Suite 0944 / Ft. Belvoir, VA 22060-6218. Phone: (703) 767-8274 or toll-free in the U.S. — 1-800 225-3842).

Table of Contents

List of Tables

Report of the STEP '97 Workshop on Net-Centric Computing

Abstract: As part of the STEP '97 conference, workshop W2b, *Net-Centric Computing,* was held on Thursday, July 17, 1997. Through focused presentations and open discussion, this workshop explored net-centric computing and its potential impact on software users, application developers, and system administrators. This report describes the STEP '97 conference, overviews the Net-Centric Computing workshop, and provides a summary of the invited presentations.

1. Introduction

STEP (Software Technology and Engineering Practice) is an international working meeting for the field of software engineering practice, drawing together practitioners and researchers concerned with software engineering's role in software and systems development, evolution, and management [Budgen 97]. The program of workshops, refereed papers, invited presentations, and tutorials allows participants to explore and report on current practical, applied experimental, and theoretical work that affects software engineering.

The STEP '97 conference was the eighth in a series of international gatherings, superseding the seven previous Computer-Aided Software Engineering (CASE) conferences. STEP '97 was held in London, UK, in July 1997; the previous conference, CASE '95 [Müller 95], was held in Toronto, Canada, in July 1995.

A report summarizing the CASE '95 workshops is available online at http://www.sei.cmu.edu/~reengineering/pubs/case95/ [Tilley 97].

STEP '97 focused on the theme of creativity and innovation in software development. The Program Committee refereed 107 papers and selected 47 for inclusion in the proceedings and presentation sessions. The conference program also contained a number of papers reporting on the introduction of software best practice, based on projects that have been funded as part of the European Commission's ESSI program. The accepted papers served as bridges to the core of STEP: the workshops. The purpose of these workshops is to foster exchange of information and ideas among researchers, developers, and users of software engineering tools and techniques. A third component of STEP is tutorials, which provide an opportunity for in-depth study of important topics from leaders in the field. STEP '97 included ten tutorials that covered topics such as analyzing patterns in data modeling, legacy system reengineering, and user-oriented software quality evaluation

In the spirit of past conferences, the STEP '97 program featured ten half-day workshops on timely and exciting topics in the field. The workshops at STEP '97 were

1. W1a: Reverse Engineering and Legacy Systems
2. W1b: Configuration Management
3. W1c: Object-Oriented Middleware, Frameworks for Systems Integration
4. W2a: What Makes for Project Success or Failure?
5. W2b: Net-Centric Computing
6. W2c: Component Based Development
7. W2d: Structured Testing
8. W3a: Choosing Techniques and Tools for Requirements Specification
9. W3b: Effectiveness of Software Process Improvement
10. W3c: Distributed Cooperative Working

More information on the STEP '97 conference can be found online at http://www.co.umist.ac.uk/STEP97. The remainder of this document will focus on Workshop W2b, Net-Centric Computing.

2. Net-Centric Computing

The explosion of interest in the Web has given rise to many new developments. One of them is Net-Centric Computing (NCC). The underlying principle behind NCC is a distributed environment where applications and data are downloaded from network servers on an as-needed basis. This is in stark contrast to the current use of powerful personal computers (PC) that rely primarily on local resources. In some respects, NCC resembles an earlier computing era of mainframes and dumb terminals. However, there are important differences. NCC relies on portable applications than run on multiple architectures (write once, run anywhere), high bandwidth (for downloading applications on demand), and low-cost "thin clients" such as the Network Computer (NC) and the NetPC for local processing power.

The adoption of NCC may have a significant impact on users, developers, and administrators. For some users, the relief at not having to maintain a PC means they can instead concentrate on their primary tasks. Other users may chafe at the limitations of NCC. The removal of "personal" from PC means users will no longer be able to significantly alter their desktop environment. However, for administrators this means a potential reduction in the cost and complexity of managing a corporate network. For developers, NCC offers an opportunity to greatly increase their customer base: Applications written in a NCC-aware programming language, such as Java, allows writing code once and having it immediately accessible on multiple platforms. This also leads to a different development environment, a new deployment model (renting applications versus buying), and new concerns about security.

Net-centric computing provides a rich source of new challenges to the software engineering community, but it is also a topic surrounded by heated debate. One of the goals of the NCC workshop was to identify short-term issues for the community and to make recommendations for further work, perhaps at the next STEP conference.

3. The Net-Centric Computing Workshop

The three-hour workshop on Net-Centric Computing was held on Thursday, July 17, 1997. The workshop leaders were Scott Tilley from the Software Engineering Institute at Carnegie Mellon University and Margaret-Anne Storey from the Department of Computer Science at the University of Victoria. The workshop's goal was to explore NCC and its potential impact on several classes of users, including end users of software applications, product developers, and system administrators—in essence, to sort out the hype from the reality.

There were approximately 22 people who attended the workshop; their names and contact information are shown in Appendix A. It was originally planned to have a commercial vendor participate in the workshop so that attendees could get a hands-on evaluation of a network computer, but that proved infeasible. Instead, the workshop focused on three themes, with invited presentations and interactive discussions structured around them.

The first theme was tutorial in nature: exchanging information about NCC and exploring issues such as total cost of ownership of network computers, the NetPC, and Java appliances. This theme was addressed by Scott Tilley's presentation, "The NCC Phenomenon," summarized in Section 4. The second theme was security in a networked environment. This theme was addressed by Moira West-Brown's presentation, "Computer Security Incident and Vulnerability Trends," summarized in Section 5. The third theme concerned leveraging investments in existing applications when moving to a net-centric platform. This theme was addressed by Kostas Kontogiannis's presentation, "Migrating Legacy Systems to NCC Environments," summarized in Section 6.

4. The NCC Phenomenon

This presentation, by Scott Tilley of the Software Engineering Institute at Carnegie Mellon University, was an overview of net-centric computing. The basis for the interest in NCC, from both the consumer and the vendor perspectives, was discussed. Various types of "thin clients" were described, including related deployment issues. It was emphasized that this is a rich area for research, as many people are currently investing heavily in studying, developing, and adopting NCC.

A motivation for the use of NCs is the high total cost of ownership (TCO) for PCs. Although most people don't care about the cost, Information Technology (IT) managers do care. This is especially true if staff arbitrarily install software and hardware on their machines. Another

motivation is that PCs rapidly become obsolete, sometimes in as little as 18 months. For some, NCC is also a form of protest against the WinTel (Microsoft and Intel) dominance in the PC world.

The underlying principle behind NCC is a distributed environment where applications and data are downloaded from network servers on an as-needed basis. This is similar to the mainframe era, except that applications are downloaded and then run locally. This style of computing relies on portable applications, high bandwidth (at least 10 Mbps, preferably 100 Mbps in a local set-up), and low-cost thin clients.

Thin clients come in a variety of guises. Some may be the more traditional display-based terminals with no local processing, while others are Java-based network computers that support local processing. A third class of thin clients is the Windows-based NetPC, which may also support Java. The developments in NCs and the NetPC are rapidly progressing. Recent developments and commercial offerings from the "Gang of Four" (IBM, Netscape, Oracle, and Sun) were discussed.

A network computer offers platform independence through Java (a distinct advantage over a personal computer), but at the same time requires that all new software is written in Java. One option for non-Java applications is to compile them to Java bytecode; this is already possible for Intermetrics Ada 95 and Parc Place/DigiTalk Smalltalk. Some legacy databases can be accessed through Java Database Connectivity (JDBC). Another option being pursued by several vendors is gearing a network computer to run other user interface sessions. For example, Insignia's NTrigue Client for Java allows you to run an NT session inside a Java applet. Obviously this can lead to a complex embedding of applications and applets, a potentially confusing "hall of mirrors."

A network computer has (at a minimum) a local processor, such as a PowerPC, no local disk, memory (at least 16M usually), and an Internet connection. The NetPC is the WinTel response to the NC: It has some characteristics of the "thinner" NC and some characteristics of the "fatter" PC (but with a sealed case that can't be opened). The NetPC uses Windows-based local processing instead of Java-based processing. It relies on Zero Administration for Windows (ZAW), NT Server, and System Development Guidelines by Microsoft. The NetPC is supposedly a stepping stone to the "Managed PC" but it is not clear what a Managed PC will be. Several vendors, including IBM, are test-marketing a NetPC.[1] It may turn out that the NetPC is a better NC than the NC [Hurwicz 97], since it supports the goals of a NC with the added benefit of running existing Windows applications. With a NetPC, you can run Java in a browser or directly in the operating system.

An interesting new development is J/Direct, soon to be available in Internet Explorer (IE) 4 and Windows 98. J/Direct provides a mechanism for Java programs to call Win32 APIs

[1] IBM has since stated that it will not market a NetPC. However, Compaq recently released their NetPC.

directly. This means an end to the "write once, run anywhere" paradigm because those programs will only run on Windows machines.

Deployment issues for thin clients are complex. Some applications and environments are more suited to thinner clients than to fatter clients. There is a perception that there are only a few examples of NCs in widespread commercial use. However, this may be due to the lack of statistics more than the lack of use of thin clients. The question of whether or not "weblications" (applications that run on network computers and use a Web browser as their primary user interface) will be successful was discussed. The network infrastructure may not be ready for widespread NCC use. Push technologies such as PointCast or those soon to be made widely available in IE4 and Netscape Communicator 4 may severely tax the bandwidth and availability requirements of the current Internet and of many intranets. An example was cited of a local area network that was down for an extended period of time because many users launched PointCast at nearly the same time upon arrival at work, thus causing an increased demand for network resources.

The presentation closed with a discussion of the implications of NCC. Users may find relief from system maintenance, but they may feel restricted by some of the limitations of current thin client technology. Developers must consider a new deployment model, a "rent versus buy" factor, which may influence the architecture of their product. The platform independence of NCC means a potentially increased customer base. Administrators may realize a reduction in TCO, but an increase in security concerns. With NCC, security should be an issue for developers and users, and not just for administrators.

5. Computer Security Incident and Vulnerability Trends

This presentation, by Moira West-Brown of the Software Engineering Institute's CERT® Coordination Center (CERT/CC) at Carnegie Mellon University, began with some history about the. The purpose of CERT/CC is to serve as a focal point for the computer security concerns of Internet users. It provides 24-hour technical support incident response. Anyone, anywhere in the world, can call the CERT/CC hotline. They won't give legal advice, or log onto your system, but they will give technical advice. CERT also issues advisories and vendor-initiated bulletins describing security risks. CERT came about due to the Internet worm attack in 1988. This worm attacked the ARPAnet (a predecessor of the Internet) and had people on both sides of the country trying to reverse engineer the worm. However, they couldn't communicate with one another because the worm was attacking their main form of communication: the Internet.

At last count the Internet consisted of over 828,000 domains (smaller networks and service providers) using a wide variety of vendor platforms and versions. Different versions of software are a big problem when dealing with security issues. Also, the diverse user

CERT® is registered in the U.S. Patent and Trademark office.

demographics of government agencies, academic and research institutions, corporate users, and home users is a problem since their diverse needs often don't align with one another.

Modern software/network intruders are prepared and organized. When CERT/CC first started, intruders exploited passwords and known vulnerabilities. Now they also attack protocol flaws; examine source files for security flaws; abuse anonymous FTP, Web servers, email, and so on. Encryption is one potential solution, but it can be abused; how to regulate encryption remains an open question. If the government has access to encryption keys, encrypted messages can be checked. This law didn't pass in US, but the UK government has access to keys that can break encryption. The tradeoff is that you can catch the perpetrators, but it is less secure.

Intruders use currently available technologies by creating easy-to-use exploitation scripts. This is a big concern for NCC: Four to five advisories were released by CERT/CC for Java in the first year that it came out. Intruders are developing sophisticated toolkits, such as "packet sniffers," which look for connections going to machines giving user names and passwords. They are increasing their impact by targeting the Internet infrastructure.

Vulnerability trends show that flaws can be found without access to source code. System call trace is commonly used, and now subroutine call traces can be used to expose flaws. The good news is that the public and vendors are becoming more security conscious, but there remains a big question about vendor liability.

Of interest to NCC is that buyers are asking of products "how quick?" and "how cheap?" but not "how secure?" Old bugs are still present in new versions, and lessons learned are not being applied. Another problem is that universities do not teach programmers how to write more secure programs. The community must assume a hostile environment, identify security risks, design with security in mind, and budget for increased cost of "defensive programming." CERT/CC is creating a taxonomy with examples and information on how to avoid vulnerabilities and is working with vendors to address these problems.

Many security issues directly affect NCC and may require changes in user perception of what is "safe," since documents now have the power to run executables. HTML documents are essentially data objects that can contain embedded programs (Java applets, scripts, or controls). There is a tradeoff between functionality and risk. Even Postscript has the potential to do damage; it is not merely a document but a programming language. Just browsing a Web page can cause code to be downloaded on the fly. There is a lack of information concerning which code is running on your browser (it is not visible), and there are few tools for users and administrators. Often, users don't realize that Java is running. Even when users are in control, they may not be aware that touching, viewing, or browsing a document can be dangerous. Although a Java applet usually runs in a sandbox environment, it could still open 50 or more windows, causing denial of service.

There is a debate as to which is preferable from a security point of view: ActiveX or Java. ActiveX basically relies on trust: you trust it completely or not at all; there is currently no other control. ActiveX relies on digital signatures for authentication. Inherent trust in another author is not good enough. Although running them in sandbox environments can increase the security of Java applications, there are only so many things you can do. It is a complex environment and it is therefore easy to make mistakes. One major problem is that you only get the "applet running" message; you have no idea what the applet is doing. It doesn't even map to operating system threads, so you can't run well-known process-monitoring programs such as "ps" either.

The complexity of NCC environments is a big threat to security. At the same time, the inherent complexity may make them more difficult to attack. In general, it is harder to identify and defend against attacks. No one should become complacent about this issue.

6. Migrating Legacy Systems to NCC Environments

This presentation, by Kostas Kontogiannis of the Department of Electrical Engineering at the University of Waterloo, highlighted several reverse engineering and reengineering issues (related to NCC) about which the research community should be aware. Migrating applications to a net-centric environment may mean developing a new architecture. Re-architecting systems can be done at two levels: macroscopic and microscopic. At the microscopic level, a scanner can be useful for extracting raw data and relations between components. At a higher level, we need to do clustering and application partitioning, and constantly analyze interactions.

Legacy code is usually full of platform-dependent code; migrating the system requires identifying and extracting this code from the system. One technique is to look for pointer arithmetic or peculiar data structures used frequently, such as overlays in data structures. This procedure can be automated by applying design recovery techniques, such as plan recognition or plan localization.

For translating old platform-dependent code to new platform-independent code, there are two approaches. The first is writing new code manually (it usually can't be done automatically). The second approach is to build wrappers. This is a good approach because migrating an entire system all at once is often too difficult and too risky. It must be decided whether existing components can be reused. Does the functionality provided by the legacy code need to be re-implemented? Is it worth it to reengineer an existing system or should a wrapper be used? The tradeoff between achieving platform independence and maintaining domain-specific platform-dependent applications must be analyzed.

Integrating a legacy system into an NCC environment requires increased attention to system interfaces and implicit environmental requirements. According to Sun, "the network is the computer." Points where user authentication is required should be identified. For

example, from Toronto you can run applications in Australia. Applications need to be integrated with tools that ensure secure communication; it is the connectivity that often creates problems.

A case study was described where PL/IX code was converted to C++ code. Reasoning's Software Refinery was used heavily for this project. One of the problems when using Refine was identifying which view to show from many different alternatives. One key problem was that there are nested procedures in PL/IX, but C++ does not support nested methods.

7. Summary

From the feedback received immediately after the workshop, it seemed that there was a lot of interest in NCC. Several of the participants were unfamiliar with many of the issues and appreciated the overview presentation of net-centric computing. There was a general feeling that since NCC is such a broad topic, a longer meeting time would have been beneficial. After the workshop, many of the participants broke into smaller informal groups to continue discussions raised during the workshop.

More information about the workshop, including presentation materials, can be found online at http://www.sei.cmu.edu/~reengineering/pubs/step97/workshop/.

Appendix A: Workshop Participants

Name	*Affiliation*	*Email*
Scott Tilley	CMU/SEI	stilley@sei.cmu.edu
Margaret-Anne Storey	UVic/SFU	mstorey@csr.uvic.ca
Kostas Kontogiannis	U. Waterloo	kostas@swen.uwaterloo.ca
Pete Collins	IBM UK Labs	pcollins@vnet.ibm.com
Dennis Smith	CMU/SEI	dbs@sei.cmu.edu
Iris Vessey	Indiana University	ivessey@indiana.edu
Cornelia Boldyreff	Univ. of Durham	Cornelia.Boldyreff@durham.ac.uk
Ali Rammal	KPMG/BarentsGroup	arammal@barents.com
Robert Clark	Stirling University	rgc@cs.stir.ac.uk
Peter Utton	BT	putton@srd.bt.co.uk
Peter Green	UMIST	peter.n.green@umist.ac.uk
Don Welch	University of MD	dwelch@cs.umd.edu
Andrea De Lucia	University of Salerno	delucia@ingbn.unisa.it
Ian Peake	Univ of Queensland	peaki@it.uq.edu.au
Steve Ingram	Lanyon Ltd	steve.ingram@lanyon.com
Elliot Chikofsky	DMR Consulting Group	e.chikofsky@computer.org
Fotis Chatzipapadopoulos	NTVA Telecomm Lab	fhatz@telecom.ntua.gr
Mike Roper	PictureTel	mroper@pictel.com
Moira West-Brown	CERT/CC SEI/CMU	mjw@cert.org
Christopher Douce	UMIST	c.douce@mac.co.umist.ac.uk
Paul Budgen	UMIST	pab@sna.co.umist.ac.uk
Toru Takeshita	Chubu University Japan	takesita@isc.chubu.ac.jp

Table 1: Workshop Participants

References

[Budgen 97] Budgen, David; Hoffnagle, Gene; and Trienekens, Jos (editors). *Proceedings of the Eighth International Workshop on Software Technology and Engineering Practice* (incorporating CASE '97). London, UK, July 14-18, 1997. Los Alamitos, CA: IEEE Computer Society Press, 1997.

[Hurwicz 97] Hurwicz, Mike. "Cheaper Computer, Part 2: PCs Strike Back." *BYTE 22*, 4 (April 1997): 66-80.

[Müller 95] Müller, Hausi A. and Norman, Ronald J. (editors). *Proceedings of the Seventh International Workshop on Computer Aided Software Engineering.* Toronto, Canada, July 10-14, 1995. Los Alamitos, CA: IEEE Computer Society Press, 1995.

[Tilley 97] Tilley, Scott R. and Wong, Kenny (editors). *Workshop Report of the Seventh International Workshop on Computer-Aided Software Engineering (CASE '95)* [online]. Available WWW: <URL: http://www.sei.cmu.edu/~reengineering/pubs/case95/> (1997).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (LEAVE BLANK)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	October 1997	Final

4. TITLE AND SUBTITLE

Report of the STEP '97 Workshop on Net-Centric Computing

5. FUNDING NUMBERS

C — F19628-95-C-0003

6. AUTHOR(S)

Scott R. Tilley and Margaret-Anne D. Storey

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER

CMU/SEI-97-SR-016

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

HQ ESC/AXS
5 Eglin Street
Hanscom AFB, MA 01731-2116

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12.A DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified/Unlimited, DTIC, NTIS

12.B DISTRIBUTION CODE

13. ABSTRACT (MAXIMUM 200 WORDS)

As part of the STEP '97 conference, workshop W2b, *Net-Centric Computing,* was held on Thursday, July 17, 1997. Through focused presentations and open discussion, this workshop explored net-centric computing and its potential impact on software users, application developers, and system administrators. This report describes the STEP '97 conference, overviews the Net-Centric Computing workshop, and provides a summary of the invited presentations.

14. SUBJECT TERMS

legacy system reengineering, net-centric computing, security, STEP, thin client

15. NUMBER OF PAGES

12

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL