SANDIA REPORT

SAND2004-0075
Unlimited Release
Printed February 2004

2003 SNL ASCI Applications Software Quality Engineering Assessment Report

Molly A. Ellis, C. Michael Williamson, Joseph R. Schofield and Lora A. Bonano

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865)576-8401
Facsimile: (865)576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.doe.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800)553-6847
Facsimile: (703)605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2004-0075
Unlimited Release
Printed February 2004

2003 SNL ASCI Applications Software Quality Engineering Assessment Report

Molly A. Ellis
Information Systems Development

C. Michael Williamson
Collaborative Environments

Joseph R. Schofield and Lora A. Bonano
Information Systems Development

Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185

Abstract

This document describes the 2003 SNL ASCI Software Quality Engineering (SQE) assessment of twenty ASCI application code teams and the results of that assessment. The purpose of this assessment was to determine code team compliance with the *Sandia National Laboratories ASCI Applications Software Quality Engineering Practices, Version 2.0* as part of an overall program assessment.

Acknowledgements

The authors would like to thank the following individuals for their reviews, comments, and contributions in preparing this document: Tania Carson, Donna Eaton, Sunita Moonka, Harvey Ogden, Todd Ritterbush, and Alex Treadway.

Table of Contents

List of Figures

List of Tables

This page left blank intentionally

Executive Summary

Introduction

This document describes the 2003 SNL ASCI Software Quality Engineering (SQE) assessment of twenty ASCI application code teams and the results of that assessment. The purpose of this assessment as specified by its sponsor, Mike McGlaun, was to

> *Determine code team compliance with the "Sandia National Laboratories ASCI Applications Software Quality Engineering Practices, Version 2.0" as part of an overall program assessment.*

The sponsor also requested that the FY03 assessment be conducted and reported such that program level results could be compared to the program level results from a similar assessment conducted during 2002. That assessment is documented in the *2002 SNL ASCI Applications Software Engineering Assessment Report* [4].

Assessment Preparation

The assessment described in this report followed the procedure described in the *Software Assessment Procedure for Sandia National Laboratories ASCI Applications SQE Practices, Version 2* [2] and utilized the gap assessment tool described in the *SNL ASCI Applications Software Quality Engineering Practices, Version 2* [3].

Prior to the beginning of the code team assessments, the assessment planning team modified the assessment procedure to reflect lessons learned from the FY02 assessment; they recruited additional assessors so that multiple three-person assessment teams could be trained and scheduled so as to accommodate the scheduling preferences of the twenty code teams; they planned and conducted SQE Practices training for all interested code teams; and they planned and conducted assessment procedure training for the ten-person assessment team.

In addition, the assessment preparation involved documenting an extended version of the scoring process used in the FY02 assessment. This scoring process became the standard used for the FY03 assessment. This scoring process started with the scoring criteria outlined in the Practices document [3] and added more defined values in order to be able to differentiate various levels of implementation. The basic '3, 2, 1' model was expanded to include '+' and '-' for each of the cardinal numbers. There was no 3+ score.

Assessment Process

The first ASCI code team assessment was conducted for the HPEMS/Xyce code team during the two days, April 30-May 1, 2003. The last ASCI code team assessment was conducted for the CUBIT/Verde code team during the two days, August 20-21, 2003. The assessment concluded on August 28, 2003 with an out-brief to the sponsor.

Each of the code team assessments consisted of an in-brief with the code team, code team interviews (a technical interview and usually a management interview), a review of the code team's objective evidence by the assessment team, scoring of the practices according to the scoring criteria in Table 3, and an out-brief to the code team to communicate/share their results.

Each assessment team consisted of an assessment lead (always one of two individuals), an experienced assessor (a participant from the FY02 assessment), and a third team member. This mix of assessors provided the potential for consistent scoring results.

Assessment Results

Two types of results are included in this report: analysis of the exit questions and analysis of the scoring of the assessment checklist. This year the agreement between the assessment sponsor and the code teams was that the assessment results would be published associating the results with the appropriate code team name. This approach was different from FY02 when the assessment results were reported anonymously. To honor the FY02 requirement, no code team names are provided when comparing FY03 results with FY02 results.

Code team results were consolidated into a single data set for purposes of analyses. Results and plots were generated both at the program and individual code team levels and include:

- program level plots representing the results for all code teams
- program level plots representing average results across all teams
- assessment-specific plots (e.g., assessment exit questions results)
- additional code team specific plots (see Appendix B)

Three representative plots are provided in this section.

During the assessments, code team members and managers were interviewed. At the end of each interview session, each interviewee was given the opportunity to identify those things that are working well, those things that are not working well, and where they would spend additional money, if available. Figure 1 represents a summary of the comments collected in answer to the question, "What is working well in your environment?"

-

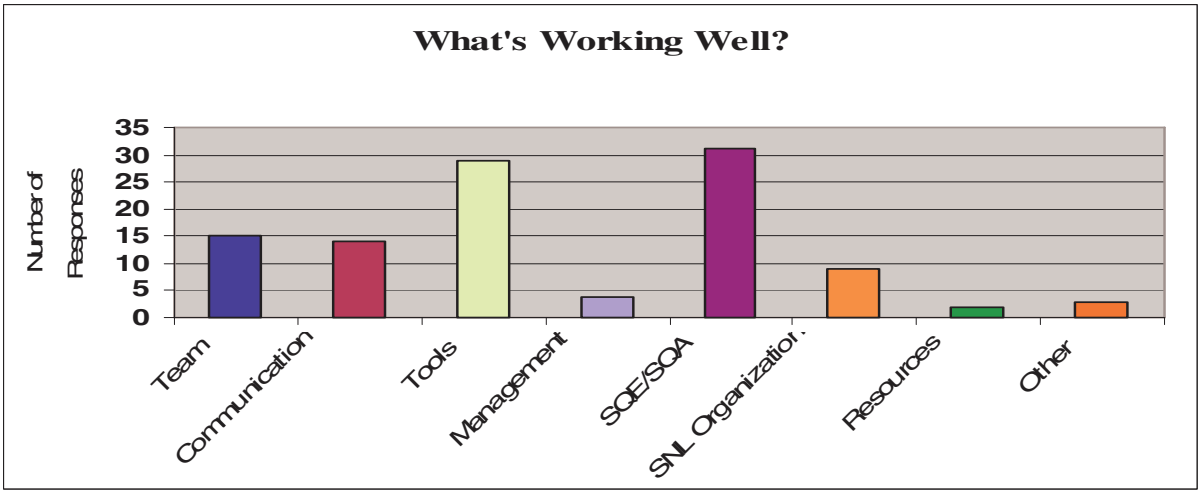


Figure 1. SQE/SQA and Tools Top the List of 'What's Working Well?'

The greatest number of endorsements related to software quality (SQE/SQA) and tools. The SQE/SQA responses focused primarily on test approaches and improvements over last year. The Tools responses included references to SourceForge, CVS, SIERRA Framework, and testing tools.

Figure 2 is a program level scatter plot that shows the sorted total normalized code team scores for both the FY02 and FY03 assessments. Data points are not paired nor are code team names provided. This plot demonstrates improvement in the overall results from FY02 to FY03. The average normalized team score improved from 91.07 percent in FY02 to 95.21 percent in FY03. Eight of the FY03 code teams attained a score of 100 percent or more of target score versus seven teams from the FY02 assessment.

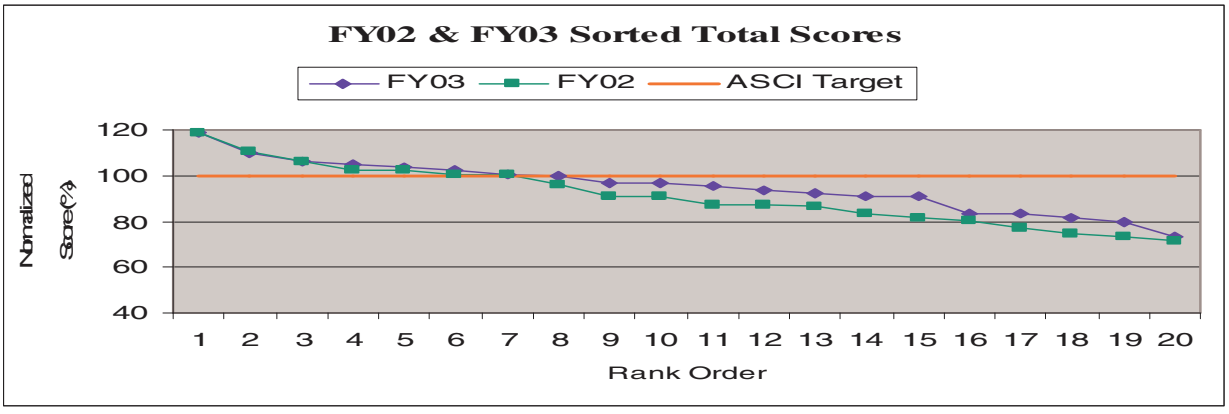


Figure 2. FY02 and FY03 Scores Follow Similar Pattern

Figure 3 presents the average assessment team score (across all code teams) compared to the ASCI management target score by practice. This plot shows the 'gap' between the desired state of the practice (as set by ASCI program management) and the current state (as determined by the assessment). This figure demonstrates where, on average, teams are exceeding the ASCI management target (represented by a green bar above the centerline). On the other hand, for practices with a red bar below the centerline, the target is not being met (i.e., further work is needed).

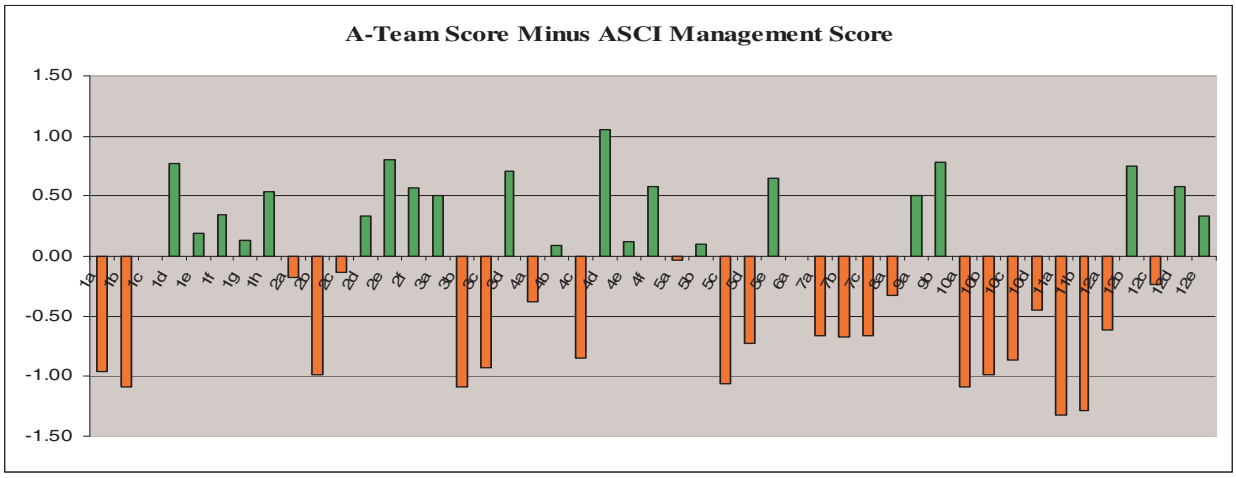


Figure 3. Gap Between ASCI Desired and Average Assessed Score

Excluding the project management practices (6a, 7a, 7b, 7c, and 8a) twenty-two practice targets are, on average, being met while twenty practice targets are not being met.

Best Practices

In addition to providing ASCI Code teams with opportunities for improvement, another purpose of this assessment was to identify best practices within the community. A best practice is based on activities that have improved quality, have improved productivity, or have enhanced customer satisfaction. A best practice is one that is used across the code team and is based on a documented process. It is often successful due to its simplicity and clarity, related training, and widespread understanding and acceptance. ***Best practices are not synonymous with world class, nor to be equated with other assessment frameworks; but they are best within the assessed community.***

These practices can serve as models for teams that desire to jumpstart improvement activity. Almost every team is cited in section 5 as having been recognized by the assessment team for one or more best practices. Table 7 (section 5) associates the ASCI software engineering practices with teams that provided adequate and documented process description and practice evidence. Of the forty-seven ASCI software practices, twenty-seven of those had at least one team that performed the practice at a best practice level. One practice, 5d, had eleven teams performing it as a best practice. Clearly practices in the areas of release and distribution demonstrate the apparent strength of the ASCI software program. In the four practices that are included in the Release Phase, 28 best practices were identified or about twenty-nine percent of all the best practices. While some teams are mentioned numerous times in the table, their

inclusion reflects an assessment score of 3- or 3, not any intended preference on the part of the assessment team.

In some cases, examples of the best practice accompany the text. While visually reinforcing, these examples are static; therefore, teams are encouraged to tap available resources, people, processes, or templates from peer projects. Teams are encouraged to leverage existing best processes. Teams are also encouraged to share documented best practices in organized forums to maximize the value of this assessment exercise.

Assessment Recommendations

This report provides two types of recommendations for improving the implementation of software quality engineering activities within the SNL ASCI applications community:

- easily implemented improvements – those activities/tasks, objectives that can be achieved with little or no cost and that can be implemented in a short time frame
- other opportunities for improvement – those activities/tasks/objectives that will require budget and/or more time and effort to accomplish.

Easily implemented recommendations discussed in section 6 include:

- document quality-improving work
- utilize available SQE resources
- document basic processes
- implement records management
- utilize an issue tracking system

Recommendations to address other opportunities for improvement, also discussed in section 6, include:

- At the ASCI program level
  - provide funding for SQE resources
  - conduct SQE training
  - provide SQE tools
  - clarify expectations of ASCI records management
  - revise the Practices document

- At the code team level
  - leverage 'like' processes
  - share SQE knowledge and expertise
  - develop formal test plans
  - identify and collect metrics

- At the assessment team level
  - revise assessment procedure
  - provide a vehicle for sharing assessment information

Section 6 also presents a number of suggested changes to the Practices document [3] based upon observations from the assessment team and upon suggestions made by many of the code teams during the assessments. Finally section 6 summarizes the lessons learned by the assessment team as a result of the FY03 assessment activities.

1 Introduction

1.1 Background

In FY01 the SNL ASCI applications code teams involved in weapons design or qualification reviewed and accepted a document, *SNL ASCI Software Quality Engineering Practices* [1]. This document describes more than forty software practices that the teams agreed to follow in order to achieve and maintain a high level of confidence in their ASCI-developed software.

During FY02, shortly after the Practices document [1] was approved and distributed, the ASCI Advanced Applications Program Manager, Mike McGlaun, sponsored a baseline software quality assessment of twenty-four ASCI application code teams. The primary purpose of that assessment was to establish the current state of software engineering practices within the Program.

The results of the FY02 assessment were documented in the *2002 SNL ASCI Applications Software Engineering Assessment Report* [4]. This report detailed the assessment results anonymously (code teams were not associated by code team name). It averaged the results across all of forty-six practices, and it charted the total scores of all assessed teams.

The FY02 assessment report [4] recommended improvements to software quality at the ASCI applications program level and identified better practices found during the assessments. It also suggested various changes that should be made to the SQE Practices. These changes were intended to clarify the meaning of some of the practices, to facilitate the reading and understanding of the practices, and to improve the assessment tool contained in the document.

As a result of the FY02 assessment, the sponsor, Mike McGlaun, requested a follow-on internal, independent assessment. This assessment would be based upon the Practices document [3] as modified and would be conducted during Q3 and Q4 of FY03. The purpose of the FY03 assessment was:

> *to determine code team compliance with the "Sandia National Laboratories ASCI Applications Software Quality Engineering Practices, Version 2.0" as part of an overall program assessment.*

The timing of the FY03 assessment was intended to give the code teams time to digest Version 2 of the Practices document [3], to read and understand the *2002 Assessment Report* [4] and its recommendations, and to implement improved processes and software quality activities. The sponsor also requested that the FY03 assessment be conducted and reported such that its program level results could be compared to the program level results from the FY02 assessment.

1.2 Scope

The FY03 assessment builds on the FY02 assessment and its recommendations. The FY03 assessment followed the process described in the *Software Assessment Procedure for Sandia National Laboratories ASCI Applications SQE Practices, Version 2.0* [2] and it utilized the gap assessment tool described in the Practices document [3].

The FY03 assessment included individual assessments of twenty different ASCI application code teams. Three of these assessments included two closely related teams. The code teams are listed in Table 1 in the order they were assessed.

Table 1. Code Teams Assessed

Code Team	Point of Contact	Dates
Xyce	Steve Wix	April 30 – May 1
PRESTO	Richard Koteras	May 14 – 15
CALORE	Ed Boucheron	May 19 – 20
FUEGO	Stefan Domino	June 2 – 3
ARIA	Sam Subia	June 2 – 3
PREMO	Curt Ober	June 4 – 5
Trilinos	Mike Heroux	June 9 – 10
Dakota	Mike Eldred	June 9 – 10
SALINAS	Garth Reese	June 11 – 12
ADAGIO/ANDANTE	Kendall Pierson	June 11 – 12
ACME	Kevin Brown	June 18 – 19
ITS	Tom Laub	June 23 – 24
CEPTRE	Jennifer Powell	June 25 – 26
NuGET	Pat Griffin	July 8 – 9
SIERRA	Kathy Aragon	July 14 – 15
Zoltan	Karen Devine	July 29 – 30
ALEGRA	Dan Carroll	August 4 – 5
NEVADA	Richard Drake	August 6 – 7
EMPHASIS/CABANA	Gary Scrivner	August 13 – 14
CUBIT/Verde	Jason Shepherd	August 20 – 21

Each code team assessment was held over the course of two days, with an out-briefing on a subsequent day. The first assessment, Xyce, was conducted April 30-May 1, 2003. The final assessment, CUBIT/Verde, was conducted August 20-21, 2003. The assessment period concluded with an out-brief to the assessment sponsor on August 28, 2003.

While code team names and scores were not made public during or after the FY02 assessment, this was not a requirement for the FY03 assessment. However, in order to protect the anonymity of the FY02 results, assessment comparisons are not made at the

code team level. Comparisons are made at the program level where summaries are generally reported.

2 Assessment Preparation

In preparation for this assessment, the assessment team began planning its approach early in FY03. Careful attention was paid to the lessons learned as published in the FY02 assessment report [4]. All of the ten lessons learned suggestions were incorporated into the FY03 approach. In particular, the assessment process [2] was revised to address those recommendations.

The first of the two-day assessments began on April 30, 2003. Prior to the first assessment significant preparation took place. This preparation began in January of 2003. It included, among many other things, planning the method for carrying out the assessment, designing an optimal schedule and scoring table, putting together an assessment team, discussions with the sponsor, making the code teams aware of the assessment, training both the code teams and the assessment team, and scheduling rooms and dates.

The first major task was to secure a qualified team of ten assessors. The project lead was responsible for first deciding how the team needed to be structured and how it was going to need to function based on the time frame for the assessments and the number of teams to be assessed. It was decided that two assessment leads would be necessary. These were chosen first due to the necessity for high levels of experience in the area of software quality. Eight other assessors were recruited on the basis of experience with software quality, a strong interest in conducting assessments, and the ability to commit to participating in multiple assessments over a four-month period.

The next task involved working with the sponsor to decide which code teams were going to participate in the assessment, and notifying those teams of their expected participation in the assessment. The teams were then asked to select a convenient assessment date and select a representative to attend the SQE Practices training. All teams were given a memorandum of understanding (MOU) to complete and sign. This MOU documented each team's assessment dates, objective evidence due dates, participating team members, and the team point-of-contact.

Training sessions for both the code teams and the assessment team were planned and developed during the preparation phase. This involved the creation of training materials and the scheduling of the sessions themselves.

2.1 Roles

The assessment planning team identified various roles and responsibilities that were necessary to accomplish the responsibilities required for the assessment. These roles and responsibilities, along with the individuals assigned to each category, are listed in Table

2. Several members of the team played more than one role and helped with tasks outside their role duties.

Table 2. Roles and Responsibilities

Role	Responsibilities	Person(s) Assigned
Sponsor	An ASCI program element manager who sponsors the internal software assessments; approves and manages the process and results; champions code teams' participation in the assessments	Mike McGlaun, 9140
Assessment Planning Team	Plan the assessment, including a review and modification of the procedure (if needed), training, scheduling, and recruiting team members	Mike Williamson, 6543 Molly Ellis, 9514 Lora Bonano, 9514 Joe Schofield, 9514 Donna Eaton, 9519
Code Team (C-Team)	Performs self-assessment, provides objective evidence; participates in interviews and activities as requested, acts on assessment results	See Table 1
Assessment Project Lead	Forms assessment teams, provides training to A-Teams as needed, plans assessment with ASCI management, plans and schedules events for all A-teams, resolves conflicts with help from assessment team leads, and reports results of assessment to sponsor	Molly Ellis, 9514
Assessment Team (A-Team) Lead	Attends required training, leads conduct of assessment as described in this document, assists the Assessment Project Lead in preparing the final report for the sponsor	Mike Williamson, 6543 Joe Schofield, 9514
Assessment Team Member	Attends required training, reviews objective evidence, participates in interviews, assists Assessment Team Lead, takes responsibility for several matrixed categories of practices in the conduct of the assessment, provides information for the out-briefing and report for the Code Teams	Mike Williamson, 6543 Harvey Ogden, 6543 Tania Carson, 9512 Todd Ritterbush, 9514 Sunita Moonka, 9514 Lora Bonano, 9514 Molly Ellis, 9514 Alex Treadway, 9514 Joe Schofield, 9514 Donna Eaton, 9519
Site Coordinator	Responsible for logistics of the assessment, assists the A-Team Lead in all interface activities with the Code Teams (this individual may come from the sponsoring organization)	Lora Bonano, Org. 9514

2.2 Scoring Process

One of the stated objectives of the assessment was to be able to compare the FY03 results to the results of the baseline assessment conducted in FY02. ASCI Management endorsed the idea of instructing the code teams to follow the scoring approach outlined in the Practices document [3]. Thus, as the code teams were trained on the assessment procedure prior to the beginning of the assessment they were instructed to score themselves according to:

- 0; indicating that the practice had not yet been included in any plans
- 1; indicating that the code team planned to implement the practice
- 2; indicating that the code team was implementing the practice and could provide some objective evidence to support that contention
- 3; indicating that the code team had fully implemented the practice

Early on in the FY02 assessment the assessment team discovered that more scoring granularity was needed to differentiate levels of implementation. The assessment team decided that a modified scoring approach would be needed in order to derive maximum value from the assessment. An addition was made to the code teams set of score values, using '+' and '-' in order to give more valuable information about the level of implementation within the numeric scores. The complete set of available scores became {0, 1-, 1, 1+, 2-, 2, 2+, 3-, 3}.

During the planning for the FY03 assessment, the assessment team modified the assessment procedure [2] to include the documented scoring table that had been approved by the sponsor and assessment leads. This allowed the assessment team members to be trained in the application of the modified scoring table and also gave the assessment team a documented criteria to refer to during the assessments. Table 3 contains the scoring criteria used during the FY03 assessement.

Table 3. Assessment Team Assessment Criteria

Score	Example
0	C-Team indicates they will not implement this practice.
1-	C-Team indicates that they will implement this practice but have no objective evidence that planning or work has started.
1	C-Team has objective evidence that planning activity has started for this practice: • meeting notes indicating that planning for this practice is being addressed (with indication of intent to complete the practice). • or, correspondence (email and other) that addresses the planning of the tasks required to implement this practice.
1+	C-Team has objective evidence that positive action has been taken on the planning for this task: • documentation of formal task assignment (with deadlines) for this practice. • or, formal schedules showing deliverables for this practice.
2-	C-Team has objective evidence that implementation has started. • preliminary drafts of either process or work products. • or, ancillary documentation (email, memos, …) of productive discussions relating to the process and/or work products for this practice.
2	C-Team has objective evidence that significant progress has been made both on the practice outputs and the process. • work products (outputs) with significant content. • and a draft practice process with significant content.
2+	C-Team has: • a final version of the work products that fully address/implement the practice. • and a final version of the process that covers this practice. • and most of the C-Team is complying with the process.
3-	All aspects of 2+ and: • C-Team has objective evidence that the practice results are repeatable and that the process has been communicated to the various stakeholders. • the work products are being shared with appropriate stakeholders. • and the process has been successfully repeated, or the process is judged by the assessors to likely be repeatable.
3	All aspects of 3- and: • the practice is at a fully implemented level (maintenance stage). The practice could be evolving, via continuous improvement, but not dramatically changing as would be the case during a prototyping. • the practice is fully integrated into the activities of the C-Team.

2.3 Training

The assessment planning team identified a need for two types of training, both of which needed to be conducted prior to the beginning of the assessment period (see Table 4).

Table 4. Training Types

Type	Description	Intended Audience	Dates
SQE Practices	• overview of the ASCI apps SQE practices • logistics of the FY03 SQE assessment • how to prepare evidence for the assessment • review of changes from FY02	• C-Teams (at least one person from each team) • A-team members (all)	• 04/02/03 • 04/07/03 • 04/08/03
Assessment Procedure	• how assessments would be conducted • roles and responsibilities of A-Team members • outline of master and daily assessment schedules • post assessment activities • preparation lab	• all A-Team members	• 04/10/03

The SQE Practices training consisted of three identical sessions, each of which lasted approximately three hours. This training centered primarily on the gap analysis tool and provided instructions on how code teams would conduct their self-assessments as well as what to expect from the internal assessment. Participants were advised on how to prepare their objective evidence, including several examples that demonstrated the difference between process evidence and implementation evidence. They were also given examples of sample interview questions that might be asked during their assessment.

Since over half of the assessment team members had not participated in the FY02 assessment, all members of the assessment team were required to attend at least one of the three SQE Practices training sessions. These training sessions were all held during the first week of April thus providing the code teams ample time to go back to their work environments and prepare for the assessments. Most of the assessments occurred one to three months after code team training. Forty-eight code team members attended one of the SQE Practices training sessions. Two other code team members were trained one-on-one by the assessment project lead.

The four-hour assessment training was attended by all ten assessment team members. This training provided some team building activities as well as background information to the team on the history of the ASCI Applications SQE practices [3], responsibilities of

each team member, how an actual assessment would be carried out, how the scoring criteria would be applied (see Table 3), the types of interview questions that would be used, and a lab on preparing good interview questions.

3 Assessment Process

The first FY03 ASCI applications code team assessment was conducted during the two days, April 30-May 1, 2003. Prior to that time a two-day activity schedule (see Table 5) was mapped out that provided adequate time for the assessment team to:

- conduct an in-brief with the code team
- conduct two code team interviews – a management interview and a technical interview
- review the objective evidence submitted by the code team
- score the practices according to the scoring criteria presented in Table 3.

Based upon these identified activities and also upon timing and budgetary constraints, the assessment planning team determined that each assessment team would consist of three people drawn from the role table (Table 2). The following guidelines were used to determine the composition of each team:

- one assessment team lead (Mike Williamson or Joe Schofield)
- at least one team member with experience from the FY02 assessment (Harvey Ogden, Alex Treadway, or Mike Williamson)
- a third team member drawn from the ten-person assessment team pool.

By applying this formula we felt that each code team profited by having an assessment team with significant experience and the potential for very consistent scoring results. Of course, by the end of the assessment period all of the other assessment team members were expected to gain significant experience.

3.1 Schedule

In order to maintain consistency in timing and organization of the twenty individual assessments, a schedule was prepared that allowed adequate time for all the planned activities involved. Shown below is a sample of the typical schedule that the assessment team followed during a two-day assessment period. The daily schedule was flexible enough to deal with code team requests and schedule conflicts.

Table 5. Sample Two-Day Assessment Schedule

Time	Day One	Day Two
8:00	A-Team Admin	A-Team Admin
8:30	In-brief	Prepare 2
9:00	Review	
9:30	Evidence	Interview 2
10:00		
10:30		
11:00	Prepare 1	Assess 2
11:30		
12:00	Lunch	
12:30		Lunch
1:00	Interview 1	
1:30		Write up
2:00		Out-brief
2:30	Assess 1	
3:00		
3:30		
4:00		

In addition to the two days of assessment activities (Table 5), each assessment team normally spent several hours prior to the actual assessment doing an evidence review. Each member of the three-person assessment team agreed to be responsible for reviewing the evidence pertaining to several areas of the practices. This review resulted in a well-prepared assessment team. If any glaring shortcomings or missing evidence sections were discovered, the code team was notified and given the opportunity to provide additional evidence prior to the start of its assessment.

The next table (Table 6) describes each of the activities that took place during the two-day assessment period.

Table 6. Assessment Activities

Activity	Time	Description
In-Brief	30 minutes	The in-brief formally began every assessment. The in-brief served several purposes: • included introductions of the assessment participants and their roles • gave an assessment overview and schedule • covered assessment conduct • allowed for initial questions
Review Evidence	2 hours	The evidence review period allowed time for the assessment team to review the evidence provided by the code team. Evidence was reviewed for all forty-seven practices and evaluated to ensure that it met the following criteria: • index existence • relevance • suitability • consistency The assessment team also looked for evidence of both process and practice for each of the forty-seven practices during this review.
Prepare (1 & 2)	1 hour	The preparation times allowed the assessment team to prepare specific questions for the code team interviews related to both the SQE practices and their review of the evidence.
Interviews (1 & 2)	1.5 hours	The assessment team conducted two interviews with each code team. These consisted of a project management interview and a technical interview. Partway through the assessment the project management practices were dropped from the assessment. After this, the two interviews were used to split up the remaining practices between the two days. This time was used to interview the code team members to assess their knowledge and understanding of their processes, to corroborate the evidence the code team had provided, and to allow for explanation of any of the practices.
Assess (1 & 2)	1.5 - 2 hours	The assessment periods were for the assessment team to spend scoring each of the practices covered during the previous interview. It also allowed the assessment team to review the interview responses from the code team and any additional evidence that might be provided.

Activity	Time	Description
Prepare Out-Brief	3 hours	This time was provided for putting together the out-brief presentation that typically included: • The code team scores for each of the 47 practices • The scoring/rating system used to score the practices • A summary comparison of the code team scores, the assessment team scores, and the ASCI management required score • A summary of findings and recommendations from the assessment team.
Out-Brief	1 hour	An out-brief was provided to each code team. It usually took place the Monday morning following the code team's assessment. The out-brief was given by the assessment team lead (or a delegate) and provided the information listed in the Write Up Out-Brief description box. It also provided a time for discussion between the assessment team lead and the code team regarding the scores given by the assessment team.

4 Assessment Results

This section addresses the results of the FY03 ASCI applications SQE assessment. Two types of results are presented: analysis of the exit questions and analysis of the scoring of the assessment checklist – the actual assessment by both the code teams and the assessment team.

4.1 Exit Questions

At the conclusion of code team assessment interviews the interviewees were given the opportunity to respond to three open-ended questions. These questions were included at the request of the sponsor and were the same as the exit questions asked during the FY02 assessment. The assessment team asked each interviewee to provide responses to the following questions:

- ***What is working well in your organization?***
- ***What is not working well in your organization?***
- ***If you had more money where would you spend it?***

The complete, unedited responses to these questions, grouped by category, can be found in Appendix C. The results are displayed in Figures 4, 5, and 6. The responses to the questions are grouped by category/theme.

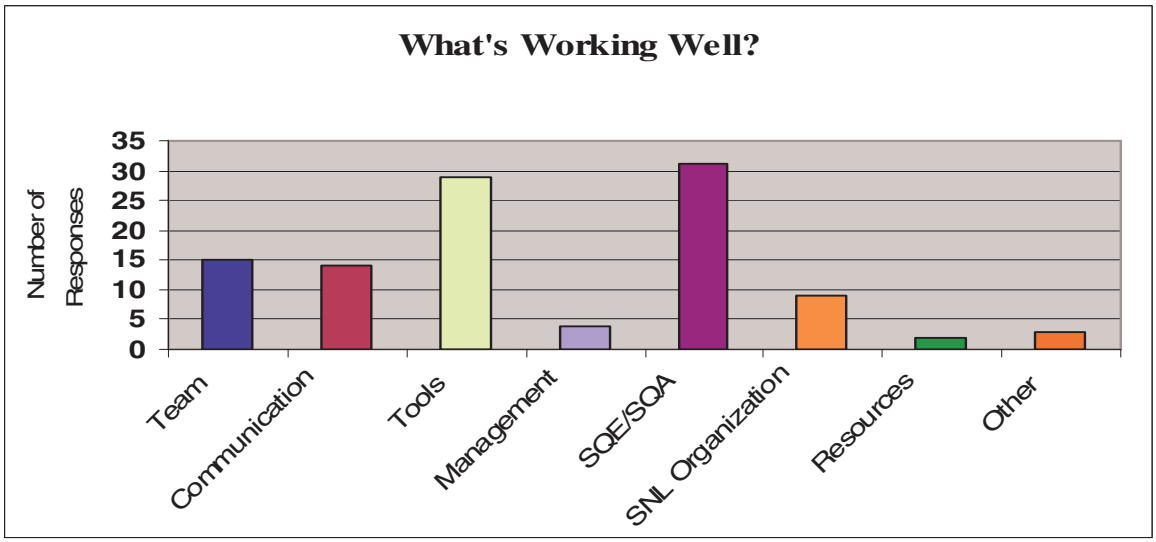


Figure 4. SQE/SQA and Tools Top the List of 'What's Working Well?'

The most common themes in Figure 4 were:

- **SQE/SQA** – test approach, improvements over last year
- **Tools** – SourceForge, CVS, SIERRA Framework, and testing
- **Teams** – small teams with good group dynamics
- **Communication** – more informal than formal, good relationships

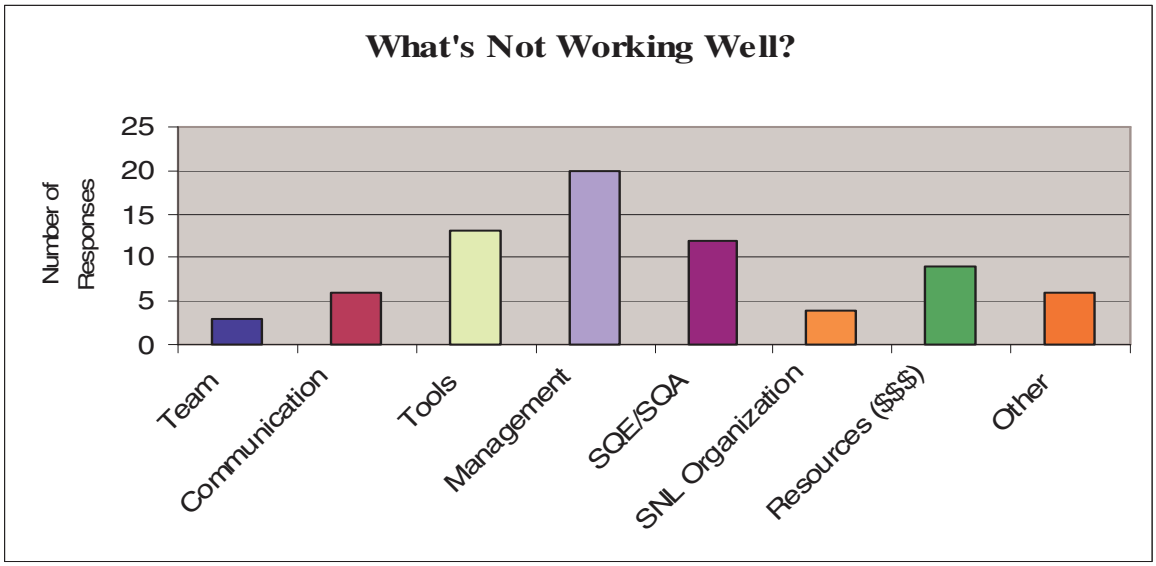


Figure 5. "Management" Most Frequently Mentioned Barrier

The most common themes in Figure 5 were:

- **Management** – changing requirements and expectations, SQA goals
- **Tools** – changing tools and platforms
- **SQE/SQA** – requirements management/tracing

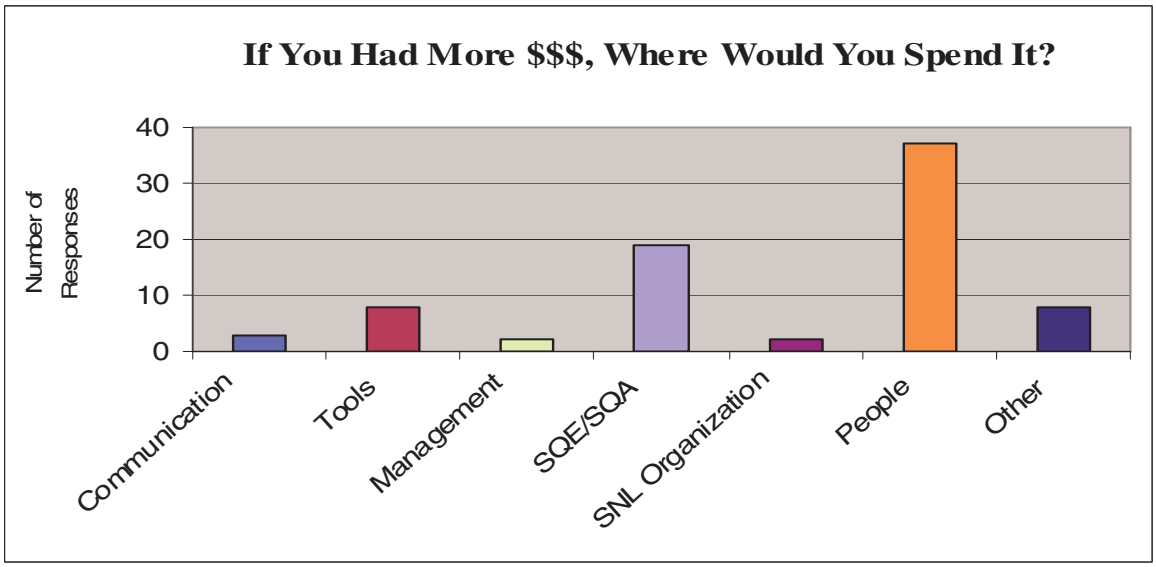


Figure 6. Code Teams Desire To Invest More In Labor/People

The most common themes in Figure 6 were:

- **People** – Hire people to do such things as document, test, code, do SQA; need more people with the 'right' skill mix
- **SQE/SQA** – Hire more people to do testing, spend more time on SQE

4.2 FY03 Assessment Results

During the FY02 ASCI Applications Software Engineering Assessment the results were recorded and reported without reference to specific code teams. Scores were published using team numbers and not team names. The anonymity was the result of an agreement between the code teams and the assessment sponsor.

Prior to the FY03 assessments the sponsor indicated that the results would be reported using code team names. In order to honor the promised anonymity of the FY02 results, no code names are provided where comparisons are made or there is some reference to FY02 data. The FY03 data, such as Appendix A, the Master Score Table, and Appendix B, which displays a page of FY03 results for each of the twenty code teams, both identify each code team by name.

It is also important to point out that while the overall FY03 ASCI Management target score was 93, the goal was subject to adjustment for each team. This adjustment was applied when code teams received a score of NA (non-applicable) on any of the project management related practices (6a, 7a, 7b, 7c, and 8a) or on the third party software practices (11a and 11b).

In FY02 the total target score set by the AQMC was 87. In FY03 ASCI management was responsible for setting the target score of 93. The difference between the FY02 and FY03 targets resulted from raising three practice scores from a one to a two and adding a new practice with a target of three. For comparison purposes both years' scores were normalized to reflect the percentage of target score achieved.

The results in this section include plots demonstrating:

- sorted total scores for FY03 and FY02 against a normalized ASCI target
- a program level bar plot showing for each practice the FY03 average assessment team score minus the ASCI Management target score
- three related plots showing
  - average FY03 and FY02 performance on the practices with a target score of 3
  - average FY03 and FY02 performance on the practices with a target score of 2
  - average FY03 and FY02 performance on the practices with a target score of 1
- a plot showing average scores by target value
- three related difference plots showing
  - difference between the assessment team score and the code team score for all twenty teams assessed based upon the FY03 assessment results
  - difference between the assessment team score and the code team score for all twenty teams assessed based upon the FY02 assessment results
  - bar chart demonstrating the cumulative difference between code team scores and assessment team scores in FY02 and FY03

Additional code team level plots are included in Appendix B.

Figure 7 is a program level scatter plot that shows the sorted total normalized code team scores for the FY03 and FY02 assessments. Data points are not paired and there is no

reference to any code team by name. The FY02 assessment included twenty-four code teams versus twenty for FY03 so only the code teams that are common to both years are included. Where two FY02 code teams were combined into one assessment in FY03, the average of those two teams is reflected in the FY02 plot.

Several conclusions can be drawn from Figure 7. The shapes of the curves for both years are very similar. The assessment team attributes this similarity to a consistent assessment process and scoring criteria from one year to the next.

The curves on this plot also demonstrate incremental improvement in the overall results from one year to the next. The average normalized team score over these twenty teams improved from 91.07 percent in FY02 to 95.21 percent in FY03. Eight of the FY03 code teams had a score of 100 percent or more versus seven teams in FY02. This improvement in FY03 scores would have been higher had practices 6a, 7a, 7b, and 7c been assessed in the same way they were in FY02. In FY02 most code teams received a '3' on these practices with the assumption that they were following the ASCI program process.

It should be noted here that the sorted results from one year to the next do not reflect the same order of teams. In fact, three of the lowest scoring teams in the FY02 assessment are now in the top five of the FY03 assessment scores. Overall, from one year to the next (FY02 to FY03) eight teams improved their scores by 5 percent or more, seven teams stayed about the same, and five teams saw their score drop by more than 5 percent.

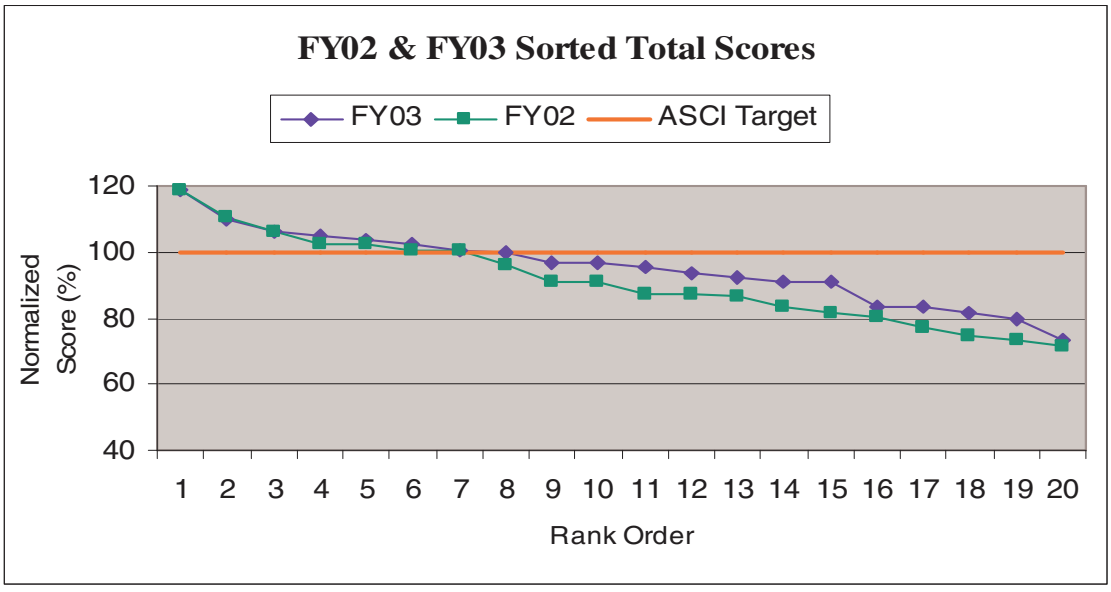


Figure 7. FY02 and FY03 Scores Follow Similar Pattern

Figure 8 shows the average assessment team score minus the ASCI management target score by practice. This plot is a useful program level measure because it demonstrates where, on average, teams are exceeding the ASCI Management targets (represented by a green bar above the centerline). On the other hand, for practices with a red bar below the centerline, the target is not being met (i.e., further work is needed).

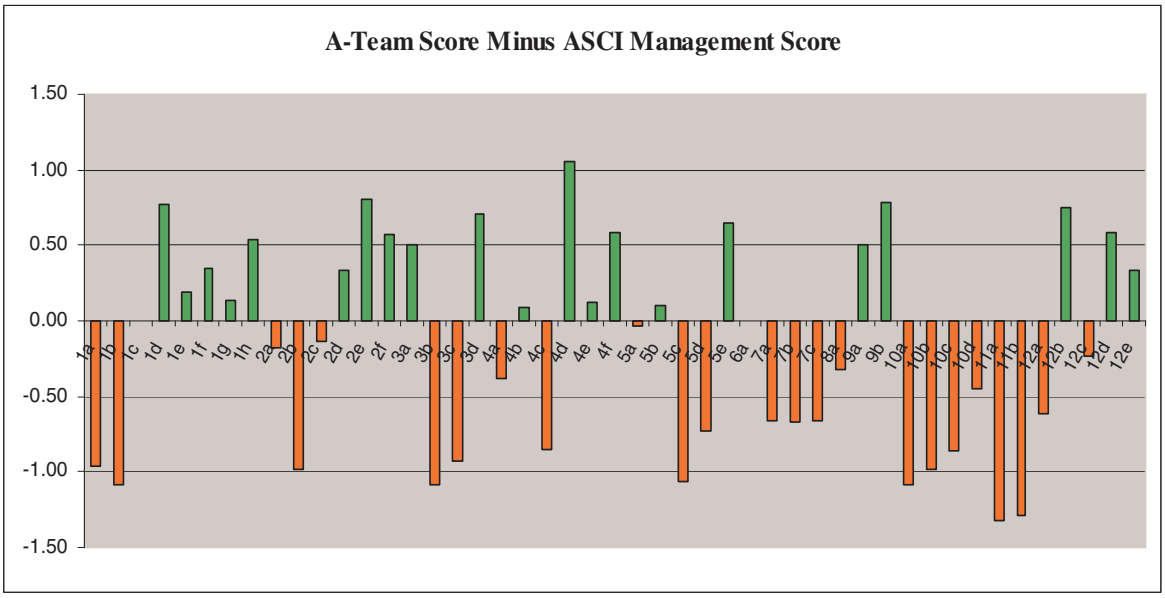


Figure 8. Gap Chart – Green Bars Exceed Practice Targets, Red Bars Indicate Work Needs To Be Done

Based upon the ASCI Management targets for the forty-two practices, excluding practices 6a, 7a, 7b, 7c, and 8a, twenty-two practice targets are, on average, being met and twenty practices are not. These numbers are consistent with the results from the FY02 assessment. Considering the same practices, in FY02 twenty-three practice targets were being met and eighteen practices were not.

The next three plots present a different view of the gaps, both for practices that have discrepancies and for practices where teams are exceeding the target. Note that for those practices targeted at a 3, the team average is less than a 2. For practices targeted at a 2, the team average is still well below the target. The practices targeted at a 1 are the only group where the team average is considerably above the target with a program level team average of 1.56.

(**Note:** FY02 Practice 1a became Practice 1b for the FY03 assessment)

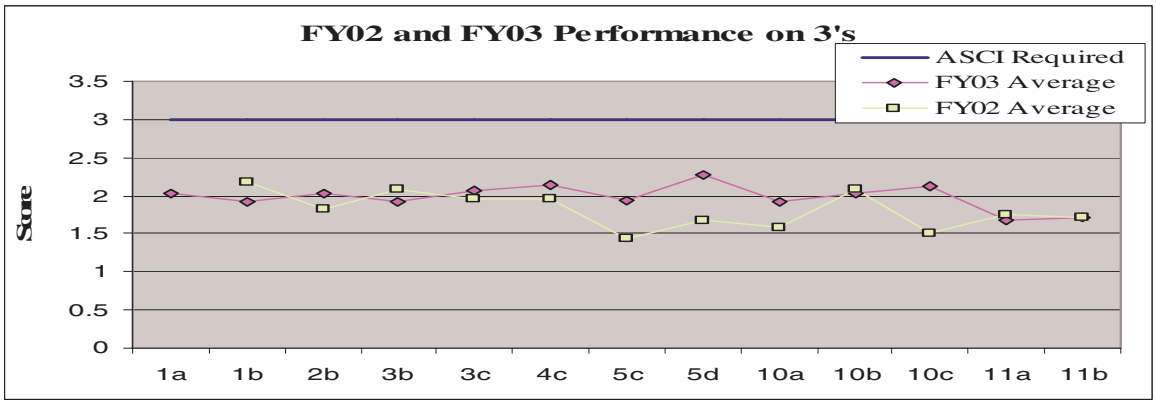


Figure 9. Teams Still Below ASCI Management Expectations For Level 3 Practices

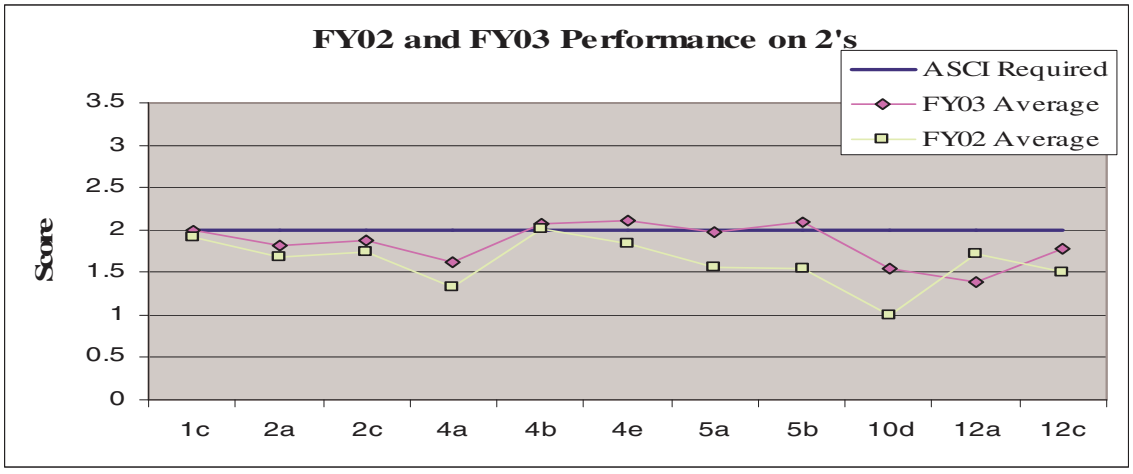


Figure 10. Performance Improvements Align Level 2 Practices With Expectations

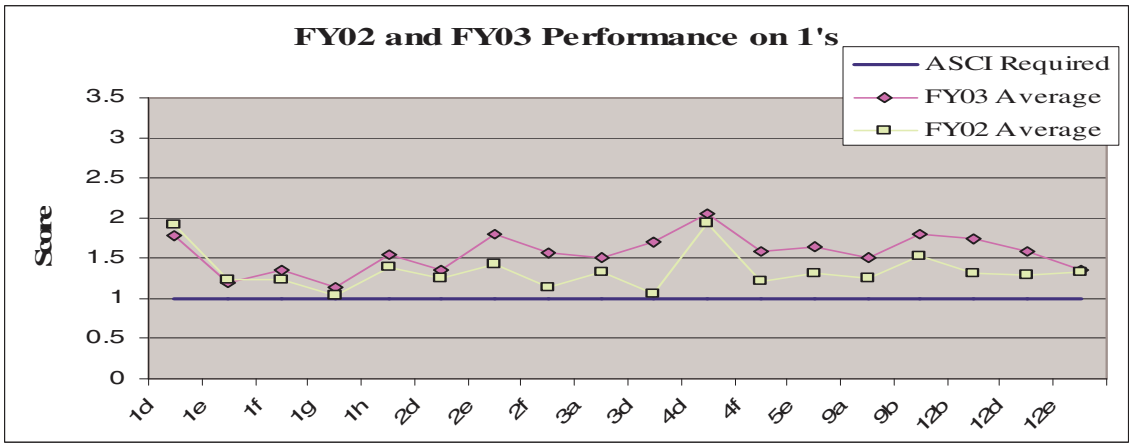


Figure 11. Performance On Level 1 Practices Exceeding Expectations

Figure 12 compares the averages achieved from the FY02 and FY03 assessments against FY03 target scores. The data sets represent averages for the practices targeted at 1, 2, and 3 respectively. The FY03 average for the forty-two practices included in this analysis is 1.79 compared to the FY02 average of 1.70 computed across 41 practices.

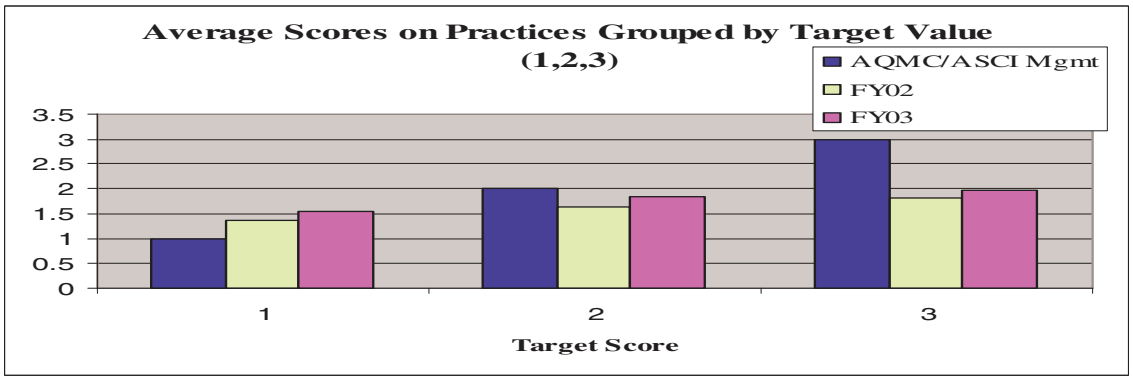


Figure 12. Averages By Target Value Show Some Improvement

(**Note:** Practices 6a, 7a, 7b, 7c, and 8a are not included in the averages shown in Figure 12.)

It is useful to examine how the assessment team scores for each team compare to the code team self-assessment scores. This comparison provides insight to the code teams' understanding of the SQE Practices document [3], code team training, and the assessment requirements. Figure 13 reflects three sets of data: the ASCI Management target (adjusted for the particular code team), the code team's self-assessed score, and the assessment team's scores of that code team. In eighteen out of nineteen cases the assessment team's scores were lower than the code team's self-assessed score.

The assessment team saw this same trend and reported it in the FY02 ASCI Apps software assessment (Figure 14).

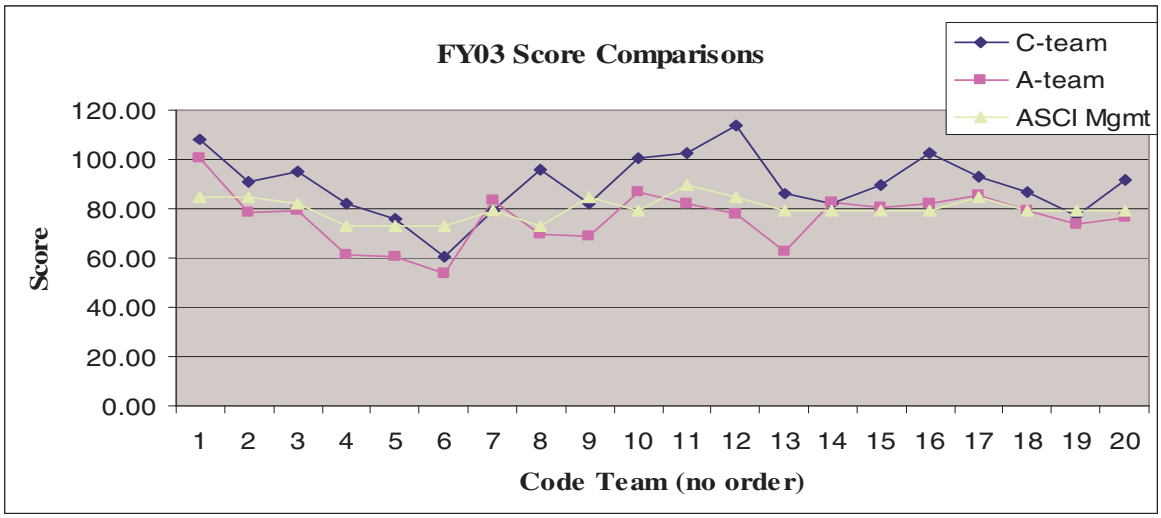


Figure 13. FY03 Code Teams Tend to Rate Themselves Higher Than Do Assessment Teams

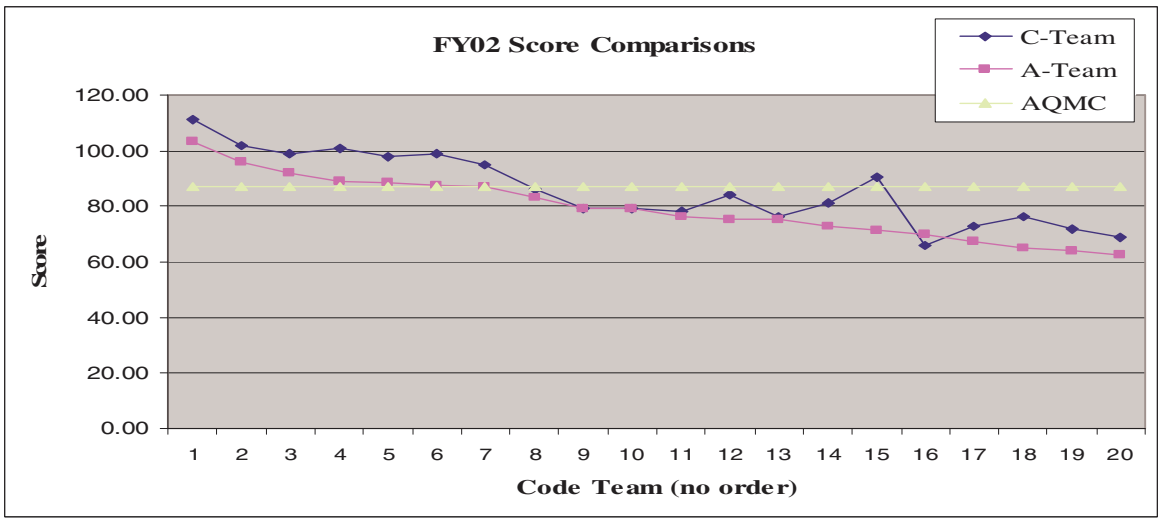


Figure 14. FY02 Code Team, Assessment Team, and ASCI Management Target Scores

Figure 15 shows the cumulative difference (the area between the C-team and A-team lines) for both years, FY02 and FY03. The cumulative difference for FY03 is almost twice that for FY02. The average difference for the twenty four code teams assessed in FY02 is 5.45; the average difference for the twenty code teams assessed in FY03 is 13.57.

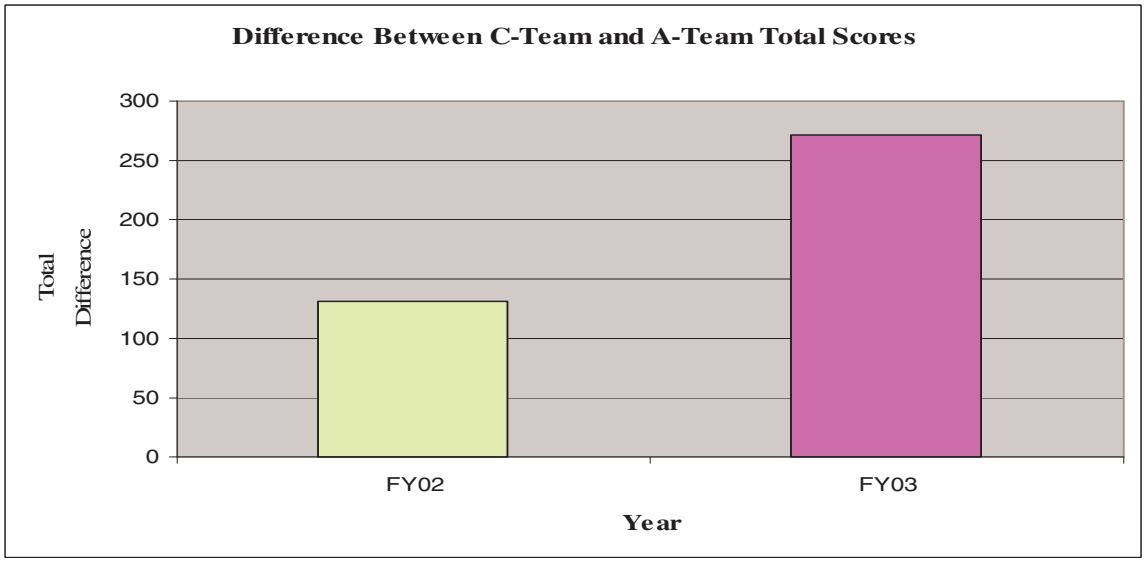


Figure 15. FY02 vs. FY03 Code Team Minus Assessment Team Difference

While it is fairly common in software assessments for the assessed team to rate themselves higher than the group who assesses, the ASCI assessment team has identified several recurring themes that they believe help to explain most of the differences. Some of the themes are common from FY02 to FY03.

- ***Misunderstanding of what constitutes process.*** Many teams presented objective evidence that supported results of how they have implemented a practice but they did not have a documented process to support the practice.

  The Practices document [3] specifies: "To be at the fully implemented level (rated as a 3), a *documented* process for the practice needs to be in place, and the team needs to be following this documented process." The assessment out-brief was modified after the first few assessments to include a slide that emphasized this point and provided some guidance to teams on what constituted a practice. One of the assessment leads, Joe Schofield, came up with the following guidance on process:

  > *a well-documented process contains inputs, outputs, roles and responsibilities, sequences and dependencies, reviews and approvals, and entry and exit criteria, as examples. A process should have many of, but not necessarily all of, these attributes. It may be textual or graphical but should not be merely imaginary or virtual.*

The scoring criteria required a documented process to achieve the higher scores. As a result, teams that presented some documented process, even if in draft form, tended to score higher on most practices than those teams that could present only ancillary artifacts, or evidence, relating to the way the practice was implemented.

- ***Code teams have smart people, but they lack SQE training and/or formal mentoring***. This theme was especially relevant to the scoring of most of the training practices. Teams consist of extremely smart people who are able to catch onto new concepts and ideas very quickly. However, the Practices document [3] specifies that project and individual training needs will be "planned and tailored in an individualized, need-based implementation." The assessment team saw very few examples of a project training plan. While many teams described informal mentoring as the way knowledge and understanding is transferred to new team members, there was very little documented that indicated who would be mentored and what information would be shared.

- ***Many teams are doing a good job of regression & release testing, but few have adequate test plans.*** Testing issues are presented in multiple sections of the Practices document [3]. As a critical component of Software Verification, testing is described in that section and five general categories of testing are introduced in section 3.2. Specific testing requirements are outlined in section 3.3.2.3 Test Sub-phase. Acceptance criteria that will be incorporated in the test plan are first introduced in section 3.3.1 Requirements Phase. The test plan or references to it are included in subsequent sections of the Software Engineering section of the document.

  While the actual work done by most code teams to support regression and release testing meets the expectations of the Practices document [3], test plans are for the most part incomplete, out of date, or inadequate. Some test plans consist mostly of a set of test cases and do not address test categories such as general testing (code coverage, memory testing, etc.), system software verification testing, and installation testing. Many plans lacked information on general test philosophy, testing, tools, schedule and frequency of tests, and a test case approval process.

- ***Teams believe 'others' are responsible for their own process.*** This theme was apparent in the way many teams perceived their role in the creation of artifacts in areas such as configuration management, requirements management, project management, and third party software. The FY03 assessment sponsor's guidance was that teams could point to other groups or entities as their means of meeting a practice but they still had to have knowledge of the process and outputs others provided.

- ***Access to SQE expertise.*** As with the FY02 assessment, code teams that had been working with an SQE consultant, possessed internal SQE expertise, or had contributed code team resources to writing the Practices document [3] scored higher than teams with limited SQE expertise.

5 Best Practices

As a result of a recommendation from the FY02 Assessment Report [4], the Practices document [3] was modified to include a definition of 'best practice':

> *Those activities that have proven to be of high value, have improved quality, have improved productivity, or have enhanced customer satisfaction. Typically, these practices are measured activities or have metrics to show their value and are leveraged across an organization.*

During the training for the FY03 assessment the code teams were advised that the assessment team would be looking for 'best practices' during the assessment. The code teams were also encouraged to nominate for best practice any processes they follow that meet the spirit of the definition.

In many code team out-briefs, the assessment lead included mention of practices or examples that the assessment team thought were exemplary. For the purposes of this assessment any practice scored at a 3- or 3 level by the assessment team was considered to be a candidate for best practice status. Only those practices that were assessed at 3- or above for at least one code team are included in Table 7.

Table 7. Practices Assessed at 3- or 3

Practice	Code Teams Assessed at 3- or 3
1a. Gather user requirements	SIERRA, Zoltan
1b. Derive software requirements	SIERRA, ADAGIO/ANDANTE, Zoltan
1c. Document software requirements	SIERRA, Xyce, Zoltan
1h. Review and approve requirements artifacts	Zoltan
2a. Derive the design	PREMO
2b. Communicate the design to the team	PRESTO, Trilinos, EMPHASIS/CABANA
2c. Document the design	EMPHASIS/CABANA
2e. Plan for testing: initiate development of test plan	FUEGO, Dakota
3b. Translate design into code and other software product artifacts	Dakota, ITS
3c. Communicate issues with requirements/design team and developers	PRESTO, ADAGIO/ANDANTE, Dakota, Xyce
4b. Execute test cases found in test plan	Xyce, Trilinos, Dakota
4c. Review test case output using acceptance criteria defined in test plan	ALEGRA, NEVADA, CUBIT/Verde, Xyce
4d. Document test case results	Xyce, PRESTO, Trilinos,

Practice	Code Teams Assessed at 3- or 3
	CUBIT/Verde
4e. Retest updated software if acceptance criteria are not satisfied	Xyce, ADAGIO/ANDANTE, ALEGRA, NEVADA, CUBIT/Verde
5a. Receive and evaluate release request	CALORE, NuGET, SIERRA, Xyce, Zoltan
5b. Plan and develop release	Xyce, SALINAS, NuGET, SIERRA, Zoltan, ALEGRA, NEVADA, CUBIT/Verde
5c. Review and approve release	SIERRA, Xyce, Dakota, Zoltan
5d. Create and distribute release	PRESTO, CALORE, ADAGIO/ANDANTE, Trilinos, Dakota, ACME, SIERRA, Zoltan, ALEGRA, NEVADA, Xyce,
9a. Conduct requirements tracing	SIERRA
9b. Determine requirements ownership and status tracking	SIERRA, Xyce
10a. Conduct issue tracking of software product artifacts, including requirements	Xyce, PRESTO, Dakota, NuGET
10b. Perform version control of software product artifacts, including requirements.	Trilinos, Dakota, Xyce,
10c. Perform release and distribution management.	Xyce, CALORE, Dakota, ACME, NuGET, ALEGRA, NEVADA
11a. Accept third party software and libraries into the application code domain.	ACME, NuGET, ALEGRA, NEVADA, Xyce
11b. Install, integrate, & control the accepted third party software.	NuGET, ALEGRA, NEVADA, Xyce, ADAGIO/ANDANTE, ACME
12b. Train staff on activities necessary for producing software artifacts.	Trilinos
12c. Train staff on use of software tools.	Trilinos

Table 8 lists the candidate best practices for each code team. As an indication of the interest in quality among these teams, almost every team had at least one example of a best practice candidate.

Table 8. Teams with Practices Assessed at 3- or 3

Code Team	Practices Assessed at 3- or 3
ACME	5d, 10c, 11a, 11b
ADAGIO/ANDANTE	1b, 4e, 5d, 11b
ALEGRA	4c, 4e, 5b, 5d, 10c, 11a, 11b
CALORE	5a, 5d, 10c
CUBIT/Verde	4c, 4d, 4e, 5b
Dakota	2e, 3b, 3c, 4b, 5c, 5d, 10a, 10b, 10c
EMPHASIS/CABANA	2b, 2c
FUEGO	2e
ITS	3b
NEVADA	4c, 4e, 5b, 5d, 10c, 11a, 11b
NuGET	10a, 10c, 11a, 11b
PREMO	2a
PRESTO	2b, 3c, 4d, 5d, 10a
SALINAS	5d
SIERRA	1a, 1b, 1c, 9a, 8b
Trilinos	2b, 4b, 4d, 5d, 10b, 12b, 12c
Xyce	1c, 3c, 4b, 4c, 4d, 4e, 5a, 5b, 5c, 5d, 9b, 10a, 10b, 10c, 11a, 11b
Zoltan	1a, 1b, 1c, 1h, 5a, 5b, 5c, 5d, 10c

5.1 Best Practices Examples

Several candidates are offered as best practices based on code team process, objective evidence, and assessment interviews. Some of the examples include several closely related practices. A best practice does not have to be complex; in fact, the simplest approach that achieves the desired results is often best.

Software tools such as SourceForge and Bugzilla are being used across many of the ASCI Code teams to provide tracking and support for issues, software defects, and incoming requirements. While these tools may imply a process in their usage, it's not a given that teams use the tools consistently within or across projects. The use of these tools is often "tailored" within projects so training on the tool and its application are essential. As an example, the PREMO team was commended for its use of SourceForge; specifically for actively using an attribute in the database to trace requirements through to the CVS versioning software.

Examples are provided of candidate best practices. Due to space limitations, only short excerpts are included.

5.1.1 Requirements Phase

The practices related to determining user requirements and to deriving software requirements provided only a few candidate best practices examples as many teams had limited evidence of their requirements process. One of the candidate best practices that applies to practices 1a and 1b was provided by the SIERRA Framework team.

1a. Gather User Requirements
1b. Derive Software Requirements

The SIERRA Framework team provided a link (http://infoserve.sandia.gov/sand_doc/2001/012560.pdf) to the *SIERRA Requirements Management Process, SAND2002-2560.* An objective of that document is

> *to establish a disciplined process whereby managers and developers of the SIERRA infrastructure and SIERRA applications share a common understanding of both the requirements specifications and how to evaluate, approve, and communicate requirement changes.*

The process defines a "Layered Set of Integrated Requirements" [section 2.5, page 14] that identifies five layers of requirements: Layer 0: Weapon Design Requirements, Layer 1: Programmatic Requirements, Layer 2: Physics and Functional Requirements, Layer 3: Modeling and Simulation Requirements, and Layer 4: Software Requirements. Layers 0 through 3 represent activities related to practice 1a. Layer 4 represents activities related to practice 1b. This document represents the "what" that needs to be done to implement requirements practices. Another document, the SIERRA Requirements Management Policy, describes how these practices will be managed.

In addition to their requirements management documents the SIERRA Framework code team provided objective evidence of requirements activities.

Fmwk_ 303	Heading	**5.7 Master Element Services**			
Fmwk_ 1005	Requirement	Shall support master element interface that allows Legendre polynomial basis functions associated with edges and faces. [SIERRA_SCR_841]	Approved	Medium	
Fmwk_ 1042	Requirement	Shall support master element interface to advertise parametric coordinate mapping, i.e. [0,1] vs. [-1,1] [SIERRA_SCR_841]	Approved	Medium	
Fmwk_ 307	Change Request	Currently SIERRA supports Nodal Interpolation, where nodes may be placed at element vertices, on element edges, and on element faces. This functionality should be extended to include basis functions associated with edges and faces, as opposed to nodes on edges and faces.	Approved	Medium	

Figure 16. SIERRA Framework Requirements Tracking With DOORS Tool

5.1.2 Release Management Phase

Example 1. The Xyce team follows a progression iteration approach, moving code from a development status through to a production status. Code modifications are controlled and testing becomes more formal as the code moves closer to production. The code team also provided objective evidence showing successful releases of their code. The following graphic contains an example from the Xyce Release and Distribution Management document. The graphic shows the Xyce promotion model and the release cycle.

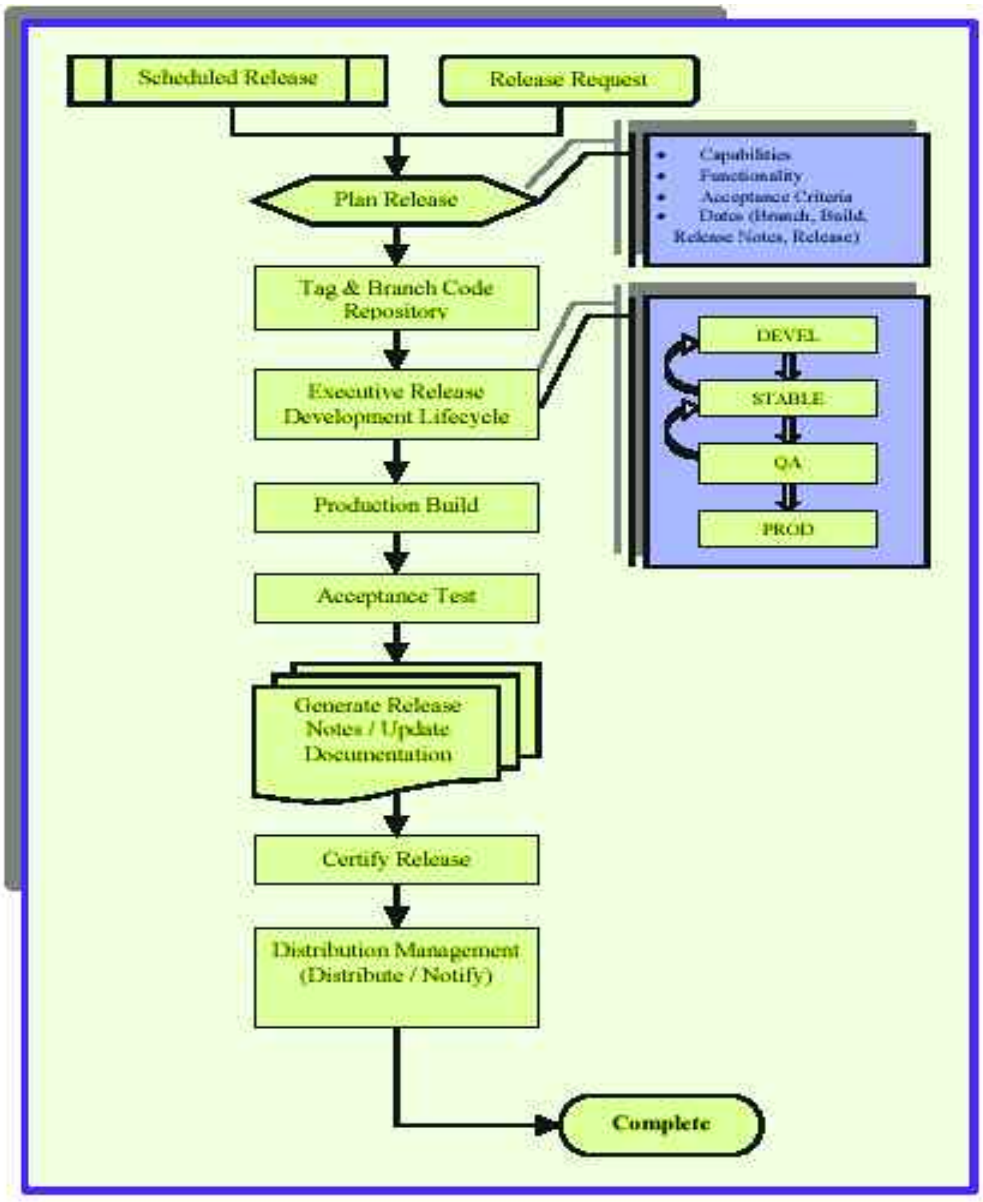


Figure 17. Xyce Release Process

Example 2. Zoltan's release checklist that addresses
5b. Plan and Develop Release
5c. Review and Approve Release
5d. Create and Distribute Release

The Zoltan team's *release checklist* for planning, developing, reviewing, approving, creating, and distributing releases includes many elements of a process. As a template the checklist ensures that consistent data is recorded for release activities, test platforms and results. The form reminds the release owner to update developer and user documentation, as well as records the date for the release of this software documentation. The completion date attribute helps to ensure the "closing" of release activities.

The Zoltan template includes general and specific directions for its use. Ownership is assigned; in this case to the project leader. Entry and exit criteria for the process, while brief, are incorporated. The sequence of task performance is implied and more specificity regarding flow could further strengthen this preferred practice.

Numerous instantiations of process evidence demonstrated that the Zoltan team had integrated the use of the template into their software engineering practice.

Finally, interview evidence suggested that the Zoltan team used the FY02 assessment results to improve their process. The use of assessment feedback for process improvement, an intent of the ASCI Code Team assessments, is itself, a best practice.

```
REQUEST

Name (responsible team member):  Karen Devine
Date Request Initiated:          03/03

Request Type:
_X_ Feature    ___ Requirement     ___ Release         ___ Bug Issue

Request Action:
___ New        _X_ Enhancement     ___ Modification    ___ Remove

Customer Priority:
_X_ High       ___ Medium          ___ Low

Source of Request (Name or Document):  Kevin Brown, PI, ACME Project

Request (Voice of Customer):

  Zoltan (mid May Delivery)
 ---------------------------
    1) Use lists for Balance call backs instead of single objects
    2) Complete robust box drop for SFC
    3) Permute the processor list to minimize data movement
    4) Maintain the cut directions from a previous balance for the
       current balance.

  See email.0302 in this directory for more details.


--  Timetable:  ACME could use enhancement ASAP; latest delivery
should be mid-May 2003.


Date of Customer Confirmation:  03/03
Tracking Date:                  03/03

Review Date:                    03/03
```

Figure 18. Zoltan's Release Request Checklist

5.1.3 *Configuration Management*

10a. Conduct Issue Tracking of Software Product Artifacts

Configuration Management is another area in which several code teams showed excellent results. For practice 10a, the Dakota team has a straightforward approach. While the Dakota process is brief, it covers the mandatory aspects of issue tracking. The following section was excerpted from the Dakota process:

> "The Dakota team has adopted the Bugzilla issue tracking system. The system is installed on a server and is used by many of the code teams. The Dakota issue tracking system is used to manage issues for all components (artifacts) of Dakota, and the user has the ability to choose which component the issue is regarding. This same system is also used to manage requirements"

An example of the Bugzilla issue tracking tool that is in use by the Dakota (and several other code teams) is included below. This open source tool captures all the essential information required for issue tracking.

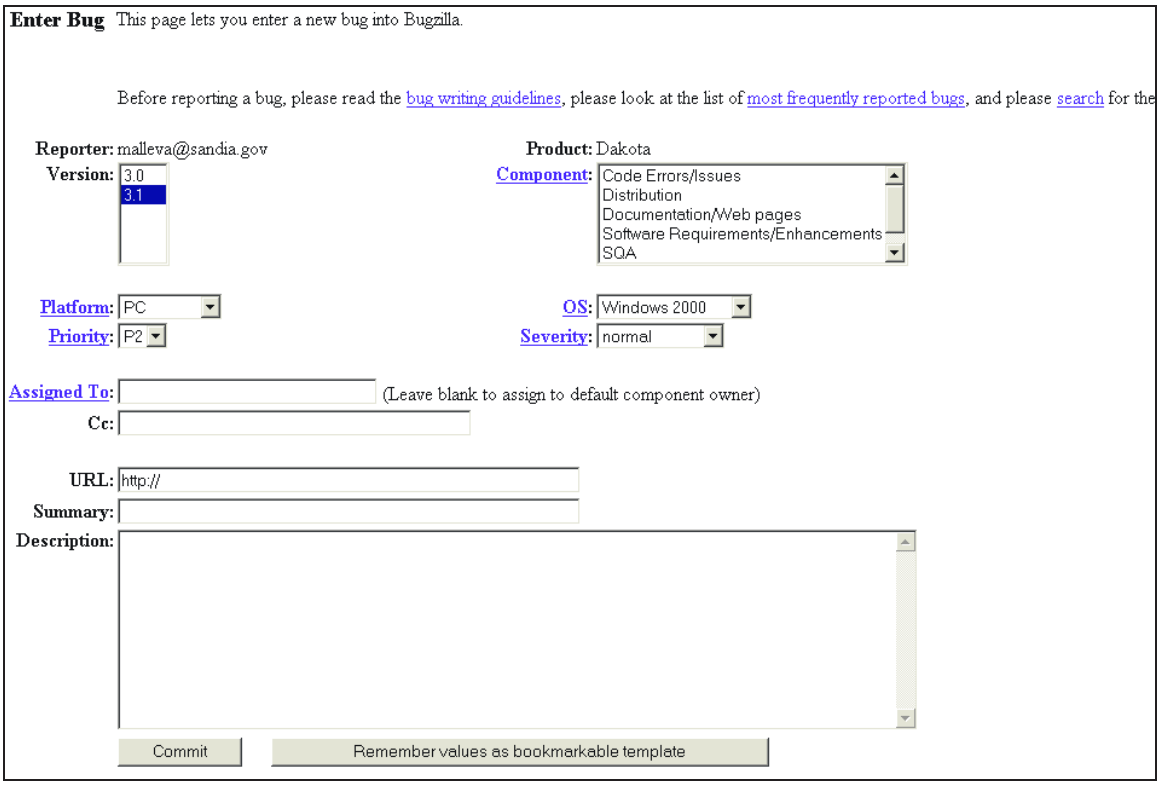


Figure 19. Dakota's Bugzilla Issue Tracking Template

The Bugzilla is not the only issue tracking tool in use by code teams. Many code teams are using SourceForge, another very good issue tracking tool, as part of their collaborative environment. All code teams that use the SIERRA Framework are either using, or intend to use, SourceForge in the near future.

5.1.4 Third Party Software Management

Not all code teams rely on third party software as an integral part of their applications but those that do must follow the practices related to accepting and integrating third party software. The Third Party Software Management area has the largest number of best practice candidates. This is not too surprising as many of these code teams support very complex environments consisting of multiple platforms, operating systems, and local configurations.

The example offered here came from the ALEGRA code team. The process contains the information required to determine "who" is responsible, "what" must be done, and "when" the activities must be completed.

The ALEGRA code team provided references to a web site that contained a detailed process and job aid for managing third party software libraries. An excerpt is provided:

Accepting a New Library

In the past few years, development activity in TPLs has increased dramatically, and these are often a source of volatility within the code. ALEGRA/NEVADA does not make use of the head of the repository for any TPL. Only released versions of the TPL will be considered for inclusion in ALEGRA/NEVADAs TPL family. The decision to include a new library version in the production version of the ALEGRA/NEVADA codes is driven by either developer or user requests. The process to verify that the TPL can be upgraded as described in the following steps. These steps are all done by the responsible party for the TPL, either the team producing it or the ALEGRA/NEVADA developer accepting responsibility for it.

1. The TPL must build on all Level 1 supported platforms.
2. The instructions given in this document concerning a new library or upgrading an existing librarymust be followed.
3. The responsible party for the TPL builds the ALEGRA/NEVADA codes with the new TPL, using special commands on the build command to force inclusion of the new library. This is done on every Level 1 supported platform. The code versions must all be able to be built.
4. The Regression Benchmark Suite tests are run on all Level 1 platforms. All Benchmarks must pass, or differences must be acceptable based upon differences in computations or capabilities introduced by the new library.
5. The responsible party for the TPL creates a new directory in the TPL directory structure described above, checks in the library files and the build scripts and modifies any Regression benchmark files that show changes with the new library.
6. The responsible party for the TPL communicates the upcoming change to the library with the development team and the users.
7. The default XML file is modified to include the new library in the production ALEGRA/NEVADA code build.

Figure 20. ALEGRA/NEVADA's Third Party Library Management Process

5.2 Other Useful Examples

5.2.1 Evidence Preparation

While not a required practice, the process of evidence collection, organization, and presentation can influence assessment results. Certainly inadequate evidence preparation can have a negative impact on the ability of the assessment team's ability to locate and evaluate the code team's objective evidence. The assessment team felt that Xyce, the first code team assessed in FY03, provided an excellent approach to evidence preparation. An example of the evidence preparation was provided to all other code teams due to be assessed in FY03. The Xyce evidence was organized, prioritized, and presented in such a way that the assessment team had no problem finding and evaluating the code team's exhibits.

The following example shows the approach to evidence collection, organization, prioritization, and presentation that was distributed to code teams.

HPEMS Xyce
FY03 ASCI Applications SQE Assessment
Comments and evidence to support the code team self-assessment – (rev 3)

NOTE: All items are numbered corresponding to the practice that they support. Evidence references generally refer to files on a CD. When provided, hardcopy evidence was labeled and included in a project binder. Each item of hardcopy evidence was uniquely numbered (starting at hardcopy item #1 – e.g. practice 1b item 5 on the next page).

The following screen print shows how we choose to structure evidence on the CD. In this example, directory 1 contains the practice sub-directories related to the requirements phase (sub-directories 1a to 1h). Following the path down, sub-directory 1a contains 8 references, numbered from 1 to 8, in order of the code teams priority. **Note:** The same reference document or link may be used in many different sections (i.e. 1a,1b, and 1c). In that case a copy of the reference (document or link) is provided in each of those directories. Finally, we choose to add a number to the beginning of the reference name to codify the intended reference priority.

Figure 21. HPEMS/Xyce Assessment Evidence Instructions

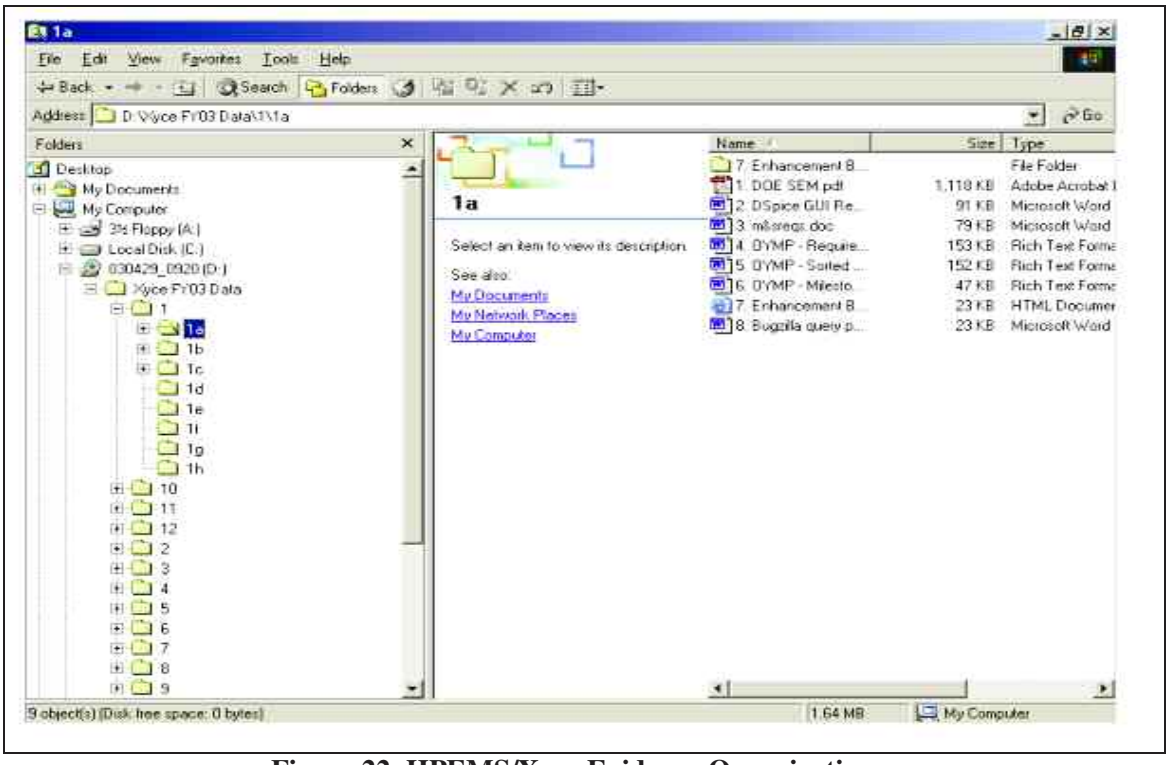


Figure 22. HPEMS/Xyce Evidence Organization

The assessment team found this approach to evidence preparation facilitated the evaluation process.

5.2.3 *Test Plan*

During the FY03 assessment no team received best practice level scores on practice 4a, Finalize Test Plan. Several teams had very good test plans that were either not complete, or not up to date. Due to the importance of a test plan, the following 'good' example of a test plan is referenced here.

Several teams had a good start on a test plan. Xyce provided a test plan that covered unit level testing, and Dakota provided a test plan that described their Agile approach to loop testing of components. No test plans presented during the assessment covered all of the aspects of testing (as described in the Practices document [3]).

As part of their objective evidence, the ITS team provided references to two documents, one containing a brief description of what is required in a test plan, and a second document titled "Verification of Analysts Requirements". Essentially this is a release test plan and while it does not address all the aspects of a complete software test plan, this document does address a significant subset including: test design, acceptance criteria, and test detail descriptions.

5.2.4 *Training*

As mentioned before, a best practice does not have to be complex. One code team, NuGET, maintains a team-member training list that shows, at a glance, the training requirements and training received by team members.

Team Member	Req. Use of NuGET Website and New Task Tracking System Database	Req. Use of MANTIS Bug Tracker	Req. E-Note NuGET Electronic Notebook	Req. CVS Training	Coverage Analysis-APROBE (vendor training)	MCNPX Intermediate Workshop	Nuclear Weapon Effects Modeling Tools Workshop	Vampir MPI Tool	Req. Use of Task Tracking System Database	SQE Inspection Process	Coverage Analysis-APROBE (SNL training)
Griffin	April 24, 2001	May 9, 2001	May 9, 2001	August 28, 2001				February 19, 2003	February 20, 2003	April 7, 2003	February 20, 2003
Parma	April 24, 2001	May 9, 2001	May 9, 2001	August 28, 2001					February 20, 2003		February 20, 2003
Cooper	April 24, 2001	May 9, 2001	May 9, 2001	August 28, 2001	February 19, 2002				February 20, 2003		February 20, 2003
Murata	April 24, 2001							February 19, 2003	February 20, 2003	April 8, 2003	February 20, 2003
Young	April 24, 2001	May 9, 2001	May 9, 2001	August 28, 2001	February 19, 2002			February 19, 2003	February 20, 2003	April 8, 2003	February 20, 2003
Vanderbeek	April 24, 2001	May 9, 2001	May 9, 2001	August 28, 2001					February 20, 2003		February 20, 2003
Bixler	April 24, 2001	May 9, 2001	May 9, 2001	August 28, 2001					February 20, 2003		February 20, 2003
DePriest				August 28, 2001	February 19, 2002	July 8-12, 2002 (Santa Fe, NM)	July 25-26, 2002 (Alexandria, VA)		February 20, 2003	April 8, 2003	February 20, 2003
Cheng	April 29, 2003	April 29, 2003	April 29, 2003	April 17,2002					February 20, 2003		February 20, 2003

Figure 23. NuGET Team Member Training Matrix

6 Assessment Recommendations

This FY03 assessment report provides two types of recommendations for improving the implementation of software quality engineering activities within the SNL ASCI applications community:

- easily implemented improvements – those activities/tasks, objectives that can be achieved with little or no cost and that can be implemented in a short time frame
- other opportunities for improvement – those activities/tasks/objectives that will require budget and/or more time and effort to accomplish.

This approach is similar to that used in the FY02 assessment report [4] and in some cases, the recommendations are very similar.

6.1 Easily Implemented Improvements

Easily implemented improvements must not require any significant cost (budget or schedule). Implementation must be quick and relatively easy to achieve. In many cases, these recommendations can be implemented by tapping infrastructure that is already in place or by making teams aware of opportunities for improvement.

6.1.1 Document Quality-Improving Work

Recommendation: The code teams should document routine activities, especially those that result in important decisions.

Discussion: documentation of activities can result in improving overall project quality. In both the FY02 and FY03 assessments, code teams often failed to provide objective evidence for activities that take place on a day-to-day basis. Examples of such activities include team meetings, hallway discussions, emails, and phone calls. Especially in the three practice areas of requirements, design, and implementation – areas in which many teams are involved in iterative code development – a shortage of evidence documenting these types of activities was especially evident.

Collecting, storing, and keeping track of periodic, decision-evident information is more than a documentation issue. It can definitely impact the overall quality of a software project. If this information is not collected and managed, it is difficult to track key decisions to completion and it is very likely that some issues will be 'lost.' Additionally, it is difficult to recall the reasoning behind certain design or implementation decisions that may affect other software development activities. This issue can be especially troublesome if the technical environment is very complex or staff turnover occurs.

Several code teams showed significant improvement in their FY03 assessment results because they used the simple and low cost solution suggested in the FY02 assessment report [4]. The assessment team saw some excellent web pages that organize and present a project's software artifacts.

This solution is re-stated here.

- Meeting discussion notes should be generated for all project-related meetings. The contents of these notes should include (at a minimum):
  - date, time, and location of the meeting
  - meeting attendees
  - important decisions and concurrence on these decisions
  - action items
- Important emails should be saved in a retrievable format
- Phone calls or hallway discussions that result in an important decision should be documented (e.g., by email) and distributed to the entire team or placed in the team's version control system
- All important project artifacts should be placed under configuration management and version control. This process can be accomplished as simply as by recording in a project notebook and then generating a printed copy, by checking the artifact into Web FileShare, or by using a formal tool with configuration management capabilities, such as SourceForge
- Project artifacts should be readily accessible to the entire team. Many teams maintain a web site that includes electronic versions of meeting notes, design notes, and other objective evidence. The method each team uses for sharing this information should be clearly communicated, understood, and used by all team members

6.1.2 Utilize SQE Resources

Recommendation: More teams should try to tap the expertise of available SQE resources.

Discussion: The FY02 and FY03 assessments both indicate that code teams that utilize a knowledgeable SQE resource (either a member of the team or access to a consultant) are implementing the SQE Practices [3] more consistently. ASCI management published a list of SNL SQE consulting resources (Appendix D) according to a recommendation in the FY02 assessment report [4]. The FY03 assessment team saw significant improvement in the results of several code teams that had been able to acquire the expertise of such SQE resources.

During the assessment several teams reported that they had budget resources to use for SQE resources but had been unable to acquire such services. Other teams reported that they did not have adequate funding to support a dedicated SQE resource, even on a part-time basis. Perhaps the answer is to share such a resource among several code teams.

6.1.3 Document Basic Processes

Recommendation: Code teams should document their basic processes that map to the forty-seven practices described in the Practices document [3].

Discussion: One of the stumbling blocks for numerous code teams in the FY03 assessment was the lack of a documented process describing how the team is

implementing the practices as described in the Practices document [3]. Teams could describe how they do it and they had examples of objective evidence to show they were carrying out a practice but they could not point to even a simple documented process that could be shared with the assessment team or with potential new members of their code team.

In order to receive a score of a 2 or higher on a practice the assessment team required a documented process. After the first few code team assessments it became evident that many teams were confused as to what a documented process entails. The assessment team has concluded that the Practices document [3] and training on its contents need to be updated to specify what constitutes a 'process' and what is expected during an assessment in terms of a documented process.

Several code teams were able to significantly improve their assessment score results during the FY03 assessment by taking some time out to write a short but meaningful description of how that team addresses the implementation of some, or all, of the forty-seven practices. As introduced in section 4.2, such documented descriptions typically include "inputs, outputs, roles and responsibilities, sequences and dependencies, reviews and approvals, and entry and exit criteria, as examples. A process should have many of, but not necessarily all of, these attributes." [Schofield]

6.1.4 Implement Records Management

Recommendation: Code teams should visit the SNL Corporate Records Management homepage to understand their responsibilities in regard to records management.

Discussion: ASCI Management raised the target score on practice 10d, Engage In Records Management', to a '2' for the FY03 assessment. While there is also a program issue as to what is expected and available for implementing records management at the ASCI project level, nevertheless, most code teams could easily improve their implementation of this practice by revisiting the SNL Corporate Records Management homepage. Records management training is required of all Sandians to give guidance on what constitutes a formal record, how to store records, what the retention period is for records, how to dispose of records, and other useful information.

In addition, the ASCI V&V program has developed a general records and document management application (RMS) to provide the capability for submittal, management, maintenance, searching, and retrieval of unclassified records, documents, and related information. While intended for ASCI V&V records, some code teams have already gained access to this system and are checking their SQE project artifacts into RMS. Other options that code teams could use, once they have determined their records management needs, are readily available ASCI tools such as SourceForge or the corporate resource, Web FileShare.

6.1.5 Utilize an Issue Tracking System

Recommendation: Code teams should investigate access to issue tracking tools.

Discussion: The FY02 assessment report recommended that code teams should have access to an issue tracking tool. Several such tools are readily available and are being utilized by various code teams. One such open source tool is Bugzilla; another commercial tool in widespread use is SourceForge. Teams that are not currently using one of these tools should give serious consideration to recording and tracking defects, enhancements, and other issues that could add value to the team.

6.2 Other Opportunities for Improvement

This section repeats many of the recommendations that were included in the FY02 assessment report [4]. These recommendations are organized according to three categories: ASCI program management, code teams, and assessment team issues. Many of these recommendations are designed to address themes introduced in section 4.2.

6.2.1 ASCI Program Management Issues

There are several opportunities for improvement in the quality program that should be addressed at the ASCI program management level.

6.2.1.1 Provide Funding For SQE Resources

Recommendation: ASCI management should fund SQE resources to code teams and they should encourage the sharing of SQE resources among those teams.

Discussion: The AQMC took a recommendation from the FY02 assessment report [4] and published a list of people (Appendix D) that could provide code teams with SQE expertise to help them in their understanding and implementation of the SQE practices. Some code teams were able to utilize individuals from this list and other teams found their own SQE expertise prior to the FY03 assessment. The FY03 assessment team observed marked improvement in those teams with access to SQE resources. However, there are still many teams that either do not have sufficient funding or the time to acquire SQE expertise.

6.2.1.2 Conduct SQE Training

Recommendation: SQE training should be conducted for all code teams at least annually.

Discussion: A reoccurring theme in both the FY02 and FY03 assessments is that code teams with exposure to SQE knowledge, such as those working with someone identified in the AQMC published SQE resource list (Appendix D), did much better in their assessment than did teams not utilizing such expertise. This exposure results in a

foundation of SQE training; teams with this foundation seem to benefit more from the SQE training that has been offered in both FY02 and FY03.

SQE training needs to target increasing the overall awareness and knowledge of the SQE practices as well as the requirements for implementing these practices. The SQE training should be expanded to include descriptions of the practice requirements, how to meet these requirements, how to document processes, and how to collect and submit adequate objective evidence. Training should also cover what resources, tools, templates, and examples are available.

6.2.1.3 Provide SQE Tools

Recommendation: ASCI program management should support the acquisition, implementation, and support of tools for use by the code teams.

Discussion: While some progress has been made since the FY02 assessment in providing SQE tools and a supporting infrastructure to the ASCI code teams, the ASCI program management needs to continue to support the acquisition, implementation, and support of tools at the program level. This recommendation will allow teams to share resources and leverage processes.

6.2.1.4 Clarify Expectations of ASCI Records Management

Recommendation: ASCI program management should provide guidance to the code teams on what they expect teams to be doing in the area of records management.

Discussion: Very few code teams assessed in FY03 understood their ASCI Records Management responsibilities. Many of the code team members interviewed expressed a desire that ASCI program management clarify expectations in this area. ASCI program management should also consider providing a program-wide tool that can be used to store and retrieve important project records.

ASCI program management has already indicated the need for code teams to at least be partially implementing records management. The ASCI V&V program has developed and implemented a records management system (RMS) that might be leveraged by all application code teams to use for storing and managing their SQE project artifacts.

6.2.1.6 Revise the Practices Document [3]

Recommendation: The ASCI Practices document [3] should be thoroughly revised in FY04 to incorporate suggested changes (section 6.3) and to map to identified industry standards.

Discussion: An approved ASCI Practices document [1 & 3] has been in existence since January 2002 and the Practices have provided the basis for two SQE assessments. The assessment team recommends that the Practices be revised. There are two major

categories of change needed. First the revision needs to bring project management practices up-to-date and to combine redundant practices (reducing the total number of practices). Second, practice descriptions should be revised to clarify tasks and activities that are needed to satisfy that practice regardless of the software methodology a team uses. Given SNL corporate awareness and focus on industry software standards, such as CMMI and ISO 9001, the Practices document [3] needs to be revised to adequately map to key areas of such standards.

Section 6.3 includes a number of specific suggested changes to the Practices document [3]. The assessment team recommends that ASCI management fund and support a thorough revision of the document in FY04. This effort should include input and recommendations from various stakeholders, including those who have planned and led the FY02 and FY03 software assessments, code team representatives who have had a chance to apply and evaluate the practices, other ASCI program element representatives, such as S&CS/OC, who have derived their own SQE practices, and those parties that will ultimately review and approve the revised Practices document [3].

6.2.2 Code Team Issues

There are several other opportunities that could well be addressed at the code team level.

6.2.2.1 Leverage 'Like' Processes

Recommendation: Look for good processes that are already implemented by other code teams and leverage these as appropriate.

Discussion: Many code teams use similar processes, for example, in the way they perform testing; in the way they prepare for a release; in the way they record and track issues; etc. Some teams have made good progress documenting these processes. Their processes are fairly mature, the processes generate good results, and the inputs, outputs, results, dependencies, etc. are well-defined and understood by code team members. Some teams have mature processes but have not taken the opportunity to document their processes. Other teams, perhaps due to the phase of development their code is in, have not addressed some practices, e.g., release area, but will need to decide on suitable processes for such practices in the near future.

Code teams should begin to share their experiences in implementing the various SQE practices. In many cases, a 'like' process can be shared from one code team to another. This has occurred where one code team member is half time on two difference projects e.g., Xyce and Dakota. Perhaps all that is necessary is for the receiving team to tailor that process to meet its own particular implementation needs.

6.2.2.2 Share SQE Knowledge and Expertise

Recommendation: Code teams should work together to establish an informal SQE practitioners working group.

Discussion: This recommendation is a direct carryover from last year. While some code teams have made great strides in their implementation of good SQE, there are other teams that are still struggling with these concepts. Perhaps the latter group has been challenged with milestones, lacks team members with appropriate SQE background, or just hasn't had time to focus on SQE.

The assessment team recommends that the ASCI application code teams establish an informal SQE practitioners working group. This group would be comprised of the SQE owners and/or practitioners from each code team plus identified SQE resources (see section 6.2.1.2). This working group would meet on a regular basis to discuss ASCI applications-related SQE issues, to share best practices, tools, templates, etc. This would provide a forum for sharing SQE practices among the code teams and would likely result in improved SQE quality. Section 6.2.2.1 addressed leveraging of processes. An applications SQE working group would be a cost-effective forum to facilitate process leveraging.

This applications SQE practitioners working group would need the appropriate level of support from ASCI program management. The group could certainly be expanded to include practitioners from other ASCI program elements.

6.2.2.3 Develop Formal Test Plans

Recommendation: Code teams should focus on developing and documenting complete and thorough test plans that incorporate the test requirements discussed in the Practices document [3].

Discussion: As introduced in section 4.2, the assessment team acknowledges that most teams are doing a good job with their regression and release testing but few have adequate test plans. Test plans presented as objective evidence consist mostly as a set of test cases and do not address the variety of test categories described in the Practices document [3]. Many plans lacked information on general test philosophy, acceptance criteria, testing, tools, schedule and frequency of tests, and test case approval process.

The test plan is first referenced early in the Practices in the Requirements Phase in practice 1e, Establish Acceptance Criteria. Practice 2e, Plan for Testing, assumes the initiation of test plan development. Test plans are vital in describing the overall verification strategy, what types of test cases are to be executed, what events trigger the execution of test cases, what defines the success of a test (acceptance criteria), and other issues that the code team deems important.

The assessment team strongly recommends that code teams make a concerted effort to develop complete and thorough test plans. See section 5.2.3 for a discussion on test plans.

6.2.2.4 Identify and Collect Metrics

Recommendation: Code teams should identify metrics that they can use to improve specific code development processes. They should then begin to collect these metrics.

Discussion: This recommendation was included in the FY02 assessment report [4] under the 'easily implemented' section. Teams may have recently begun attempts to identify metrics as described in Metrics for the ASCI Advanced Applications Program, a set of guidelines from the application program manager. However, during the FY03 assessment, the assessment team saw little or no evidence that code teams have addressed the issue of metric identification and collection. Perhaps it is not as easily implemented as originally thought. This report includes this recommendation again because it is generally recognized that any process improvement must be based upon sound metrics. Metrics provide an indication of what is working and what is not. They may also give an indication of what aspects of a process or a procedure add value and what aspects do not.

Metrics are a necessary part of process improvement. Metrics can be used to provide insight into the 'goodness' of software products and of the SQE practices used to develop the software products. Metrics provide insight into what is needed and when it is needed.

A prerequisite to generation of metrics is the collection of metric data. Collection requires that teams have the ability and the need to collect requisite data. There are numerous potential metrics that might be of value to the code teams and to the ASCI program managers. Examples include estimated vs. actual effort for development of a particular set of requirements, projected budget vs. actual budget required, number of defects reported during a specified time period vs. number of defects resolved during the same period. However, prior to collecting data for metrics, it is essential that a decision be made as to what metrics are valuable to the project and how those metrics will be used.

6.2.3 Assessment Team Issues

The FY03 assessment team consisted of ten individuals who represented various SQE and assessment disciplines. Needless to say, the assessment team pool will probably experience turnover before another ASCI applications program level assessment is conducted. There are several issues that the assessment team should address.

6.2.3.1 Revise Assessment Procedure

Recommendation: This assessment team recommends to its successors that they take a careful look at the assessment procedure [2] and revise it according to recommendations listed in section 6.4 of this report.

Discussion: See section 6.4.

6.2.3.2 Provide a Vehicle For Sharing Assessment Information

Recommendation: The ASCI assessment team should be kept at least partially intact. It should identify a vehicle for promoting and sharing ASCI SQE assessment information.

Discussion. With ASCI program management endorsement and funding, the assessment team should consider developing an ASCI SQE Assessment web page. This page could be used to share various assessment concepts, schedules, guidelines, etc. It could also be used to provide best practice examples and templates for code teams that are just starting their SQE journey or for other teams that are looking for ways to improve their practice implementations.

6.3 Suggested Changes to Practices Document [3]

After spending forty days with twenty different code teams examining their understanding and implementation of the ASCI SQE practices, the assessment team feels strongly that the Practices document [3] needs revision. As opposed to last year, when only minor changes were recommended, this year the recommendations are more extensive. The recommendations that are included here reflect not only the opinion of the assessment team but also reflect suggestions that were made by many of the code teams. Both groups recognize that there is redundancy, confusion, and inconsistency in the way some of the practices are organized and presented.

The suggested changes include:

- ***Revisit the three development area sub-phases*** – some of these practices need rewording in order to be relevant to teams that are following Agile software development methods. This suggested change is one of the most critical recommendations.

- ***Provide a definition of 'process' and what is expected to be included in the description code teams present.*** This suggestion has been referenced in multiple sections of this report and was identified as a severe shortcoming during this year's assessment activities.

- ***Revisit the three project management practices – they are all out-of-date.*** The procedure for submitting annual implementation plans, the requirement for doing quarterly reviews and for submitting baseline change proposals, and the guidelines for identifying risk all need to be reviewed and modified. Also, in the opinion of the assessment team, the entire project management section is weak in the amount of rigor it requires of teams responsible for producing quality software. Compared to other industry standards, the practices in this area are lacking rigor and completeness and should be enhanced.

- ***Consider combining the requirements management practices (area 9) with the requirements phase (area 1) practices.*** In the original development of the Practices document [3], requirements management was treated as a support element of requirements gathering and derivation and was described in a separate section. However, in both the FY02 and FY03 assessments, most code teams wound up addressing practices 1f, 1g, 9a, and 9b as though they were the same practices.

- ***Consider combining practice 10c Perform Release and Distribution Management with the release management practice 5d Create and Distribute the Release.*** In the original development of the Practices document [1], performing release and distribution management was considered to be part of configuration management and it included concepts such as baselining and promotion. However, there is a great deal of overlapping description between these two practices and they could probably be combined as one. Baselining and promotion should be retained as part of the practice.

- ***Consider revising the assessment checklist to be two-dimensional: a column for process and a column for work products supporting the process.*** The FY03 assessment team was very careful to examine code team submissions for evidence of both process and results. The majority of teams had some evidence of work product results and some teams had outstanding evidence of work product results.

  The Practices document [3] specifies that at least a draft of a documented process must be in place in order to achieve scores in the '2' and '3' range. However, many code teams had limited objective evidence of a documented process. The lack of documented process is partially due to code teams lacking a definition of what constitutes a process. The scoring instructions in the Practice document [3] and the scoring table identify only a single dimension that includes process and work products. A code team with significant or complete results sometimes received combined scores on process/results that were lower than a single score on results might have been. By going to a 2-dimensional assessment checklist, both code teams and the assessment team could give a clearer indication of code team strengths and weaknesses.

- ***Clarify forward/backward practice linkage issues.*** There are three practices in the software engineering development phases that attempt to establish or address linkages to other practices:
  - 1f – Determine necessary links to other layers of requirements, code, and tests
  - 2d – Evaluate impact to requirements
  - 3a – Evaluate impact of implementation to design and requirements

  The assessment team found that many code teams were unclear as to the value and intent of these three practices. These practices need to be clarified so that the function and value of forward/backward linkages is apparent.

  Forward linkages facilitate the tracing of requirements and design decisions through implementation and test facilitating software verification activities (e.g. requirement

and design based testing). Forward link practices also encourage a systematic approach to software engineering. Backward links facilitate traceability and maintainability (i.e. provide the ability to select a code feature or test case and determine what requirements or design attributes will be impacted by changing the code or test case).

- ***Consider consolidating some of the training practices.*** Out of the forty-seven practices on the assessment checklist, five of these currently pertain to training. The Practices document [3] contains only two short paragraphs that address those five practices. The assessment team feels that the five training practices make up an inordinate percentage of the total practices being considered and evaluated. In addition, both the code teams and the assessment team experienced difficulty differentiating the training practices 12b, 12c, and 12d.

- ***Address issues associated with Agile methodologies.*** The seven approaches to software engineering that are known as Agile methodologies span a significant range of organization, complexity, and formality.

  Most of the Agile methodologies are refinements of iterative or spiral approaches to software engineering, and align (to some degree) with the phases, practices, and outputs identified in the Practices document [3]. However, the Agile method known as Extreme Programming is interpreted by some code teams as diverging considerably from the phases, practices, and outputs defined in the Practices document [3].

  While the Practices document [3] indicates an intent to be methodology neutral, the assessors found it difficult to use the Practices document [3] to evaluate code teams following an Extreme Programming approach.

  The assessment team recommends that the Practices document [3] be revised so that mapping to various Agile methods can be more easily accomplished. The assessment team also recommends that the Practices document [3] identify the minimal set of product artifacts that must be provided by all projects without regard to the software development methodology the code team follows.

6.4 Lessons Learned

The FY02 assessment report [4] also included a section on the lessons learned from the assessment conducted in 2002. During the planning phase for this FY03 assessment, the assessment team incorporated each and every lesson learned to some degree. As a result, the problematic issues that surfaced last year were greatly reduced and the process was greatly improved overall.

At the conclusion of the FY03 code team assessments, the assessment team held another session to identify lessons learned. The main focus of this session was to review feedback from code teams and to discuss various aspects of the assessment activities from the perspective of that team. This section documents the lessons learned (and verified) during the FY03 assessment.

1. Management support is critical. In lieu of management presence at every training session, a videotape of management commitment should be shared with code teams.
2. A two-day schedule works well for individual code team assessments and allows ample time for interviews, evidence review, and scoring of the practices.
3. Assessment leads need more time to identify and write up best practice examples during the course of the assessments.
4. Training is vital for sharing the details of how the assessment will be conducted and what is expected of code teams. Future training sessions need to include more best practices and examples of what the assessment team will be looking for. A pre-assessment management training session should be scheduled. A post-assessment session is needed to share best practices from the assessment thereby helping teams initiate improvements.
5. A three-person assessment team worked very well for assessing individual code teams. Having an experienced knowledgeable lead assessor and others with subject matter expertise is also critical.
6. Having a large assessment team pool provides flexibility and allows reassignments when scheduling conflicts arise. If assessments are to be conducted over an extended period this flexibility is important.
7. Assessment out-briefs were well-received and provided valuable feedback to code teams. More time should be dedicated to preparation and delivery of these out-briefs.
8. The technical interview should probably occur on day one of the assessment followed by the project management interview on day two. The afternoon of day two should be devoted to scoring and out-brief preparation.
9. The code team's evidence must be well organized and indexed in order for the assessment team to be able to do its job efficiently and effectively in the two-day assessment timeframe.
10. In-briefs to code teams that have participated in prior assessments have limited value. The in-briefs should be reduced to ten minutes and combined with the first interview.

11. The site coordinator role is essential to a well-organized and well-executed assignment. Assessors do not have the time to deal with issues related to schedule or evidence collection.

7 References

1. Zepper, John, Aragon, Kathy, Ellis, Molly, Eaton, Donna, Byle, Kathleen, *ASCI Applications Software Quality Engineering Practices,* SAND2002-0121, Sandia National Laboratories, January 2002.
2. Ellis, Molly, Schofield, Joe, Eaton, Donna, Williamson, Mike, *Software Assessment Procedure for Sandia National Laboratories ASCI Applications SQE Practices, Version 2,* WFS080379, Sandia National Labs, January 2003.
3. Zepper, John, Aragon, Kathy, Ellis, Molly, Eaton, Donna, Byle, Kathleen, *ASCI Applications Software Quality Engineering Practices, Version 2,* SAND2003-0962, Sandia National Laboratories, April 2003.
4. Williamson, C. Michael, Ogden, Harvey C., Byle, Kathleen, *2002 SNL ASCI Applications Software Engineering Assessment Report,* SAND2002-2064, Sandia National Laboratories, July 2002.

Acronyms

A-team	assessment team
AQMC	ASCI Quality Management Council
ASCI	Accelerated Strategic Computing Initiative
C-team	code team
CMMI	Capability Maturity Model Integration
HPEMS	High Performance Electrical Modeling and Simulation
ISO	International Organization for Standardization
MOU	Memo of Understanding
RMS	Records Management System
S&CS/OC	Simulation and Computer Science/On-going Computing
Sandia	Sandia National Laboratories
SQA	software quality assurance
SNL	Sandia National Laboratories
SQE	software quality engineering
TPL	third party library
V&V	Verification and Validation

Appendix A Master Score Table

		HPEMS/Xyce				PRESTO				CALORE				FUEGO				PREMO			
		ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP
1	1a	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.33	-0.67	3.00	3.00	1.67	-1.33	3.00	3.00	2.33	-0.67
2	1b	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33
3	1c	2.00	3.00	3.00	1.00	2.00	2.00	2.67	0.67	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	3.00	1.67	-0.33
4	1d	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00	1.00	3.00	1.67	0.67	1.00	2.00	1.67	0.67
5	1e	1.00	2.00	2.00	1.00	1.00	1.00	0.67	-0.33	1.00	2.00	0.67	-0.33	1.00	2.00	1.00	0.00	1.00	2.00	2.00	1.00
6	1f	1.00	2.00	1.33	0.33	1.00	2.00	2.00	1.00	1.00	2.00	1.67	0.67	1.00	2.00	1.33	0.33	1.00	1.00	1.00	0.00
7	1g	1.00	2.00	1.33	0.33	1.00	2.00	2.00	1.00	1.00	1.00	0.67	-0.33	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00
8	1h	1.00	2.00	2.00	1.00	1.00	1.00	1.33	0.33	1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67
9	2a	2.00	3.00	2.00	0.00	2.00	2.00	2.33	0.33	2.00	3.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	3.00	2.67	0.67
10	2b	3.00	3.00	2.00	-1.00	3.00	3.00	3.00	0.00	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	3.00	2.33	-0.67
11	2c	2.00	3.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	3.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	3.00	2.00	0.00
12	2d	1.00	2.00	2.00	1.00	1.00	1.00	0.67	-0.33	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00
13	2e	1.00	2.00	2.33	1.33	1.00	1.00	1.00	0.00	1.00	3.00	2.00	1.00	1.00	3.00	2.67	1.67	1.00	2.00	1.67	0.67
14	2f	1.00	2.00	2.00	1.00	1.00	1.00	0.67	-0.33	1.00	2.00	1.33	0.33	1.00	2.00	1.00	0.00	1.00	2.00	1.67	0.67
15	3a	1.00	2.00	2.00	1.00	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00
16	3b	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00
17	3c	3.00	2.00	2.67	-0.33	3.00	3.00	3.00	0.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33
18	3d	1.00	2.00	2.00	1.00	1.00	1.00	0.67	-0.33	1.00	2.00	1.33	0.33	1.00	1.00	1.00	0.00	1.00	3.00	1.67	0.67
19	4a	2.00	2.00	2.00	0.00	2.00	2.00	1.00	-1.00	2.00	2.00	1.67	-0.33	2.00	1.00	1.33	-0.67	2.00	2.00	1.00	-1.00
20	4b	2.00	3.00	3.00	1.00	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	3.00	2.00	0.00	2.00	3.00	2.00	0.00
21	4c	3.00	3.00	3.00	0.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00
22	4d	1.00	2.00	3.00	2.00	1.00	3.00	3.00	2.00	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00
23	4e	2.00	3.00	3.00	1.00	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	3.00	2.00	0.00
24	4f	1.00	2.00	2.33	1.33	1.00	2.00	2.00	1.00	1.00	1.00	1.67	0.67	1.00	1.00	1.33	0.33	1.00	1.00	1.00	0.00
25	5a	2.00	3.00	3.00	1.00	2.00	2.00	2.00	0.00	2.00	2.00	3.00	1.00	2.00	2.00	1.67	-0.33	2.00	1.00	1.00	-1.00
26	5b	2.00	3.00	3.00	1.00	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	1.00	1.00	-1.00
27	5c	3.00	3.00	3.00	0.00	3.00	2.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	1.00	1.00	-2.00	3.00	1.00	1.00	-2.00
28	5d	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00	3.00	1.00	1.00	-2.00	3.00	1.00	1.00	-2.00
29	5e	1.00	1.00	2.00	1.00	1.00	1.00	1.67	0.67	1.00	2.00	2.00	1.00	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00
30	6a	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
31	7a	3.00	3.00	3.00	0.00	3.00	3.00	2.00	-1.00	3.00	3.00	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
32	7b	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
33	7c	3.00	3.00	3.00	0.00	3.00	3.00	2.00	-1.00	3.00	3.00	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
34	8a	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
35	9a	1.00	2.00	1.33	0.33	1.00	2.00	2.00	1.00	1.00	1.00	1.00	0.00	1.00	2.00	1.00	0.00	1.00	1.00	1.00	0.00
36	9b	1.00	3.00	2.67	1.67	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00	1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00
37	10a	3.00	3.00	2.67	-0.33	3.00	3.00	3.00	0.00	3.00	2.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	1.00	1.00	-2.00
38	10b	3.00	3.00	2.67	-0.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33
39	10c	3.00	3.00	3.00	0.00	3.00	2.00	2.00	-1.00	3.00	2.00	3.00	0.00	3.00	2.00	2.00	-1.00	3.00	1.00	1.00	-2.00
40	10d	2.00	2.00	2.00	0.00	2.00	1.00	1.00	-1.00	2.00	2.00	1.00	-1.00	2.00	1.00	1.00	-1.00	2.00	2.00	2.00	0.00
41	11a	3.00	2.00	2.67	-0.33	3.00	3.00	1.67	-1.33	3.00	0.00	1.33	-1.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
42	11b	3.00	3.00	2.67	-0.33	3.00	3.00	1.67	-1.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
43	12a	2.00	2.00	1.67	-0.33	2.00	2.00	1.00	-1.00	2.00	2.00	2.00	0.00	2.00	2.00	0.67	-1.33	2.00	2.00	0.67	-1.33
44	12b	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67
45	12c	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33
46	12d	1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	2.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67
47	12e	1.00	2.00	1.67	0.67	1.00	1.00	0.67	-0.33	1.00	2.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67
Totals		85.00	108.00	101.02	16.02	85.00	91.00	78.37	-6.63	82.00	95.00	79.03	-2.97	73.00	82.00	61.06	-11.94	73.00	76.00	60.71	-12.29
% of		100.00	127.06	118.85	18.85	100.00	107.06	92.20	-7.80	100.00	115.85	96.38	-3.62	100.00	112.33	83.64	-16.36	100.00	104.11	83.16	-16.84

ARIA				Trilinos				SALINAS				Dakota				ADAGIO/ANDANTE			
ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP
3.00	2.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00
3.00	2.00	1.00	-2.00	3.00	3.00	1.67	-1.33	3.00	2.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	3.00	2.67	-0.33
2.00	2.00	1.33	-0.67	2.00	2.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	3.00	2.00	0.00
1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00
1.00	2.00	0.67	-0.33	1.00	2.00	1.00	0.00	1.00	1.00	1.67	0.67	1.00	2.00	1.00	0.00	1.00	2.00	1.67	0.67
1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	1.00	0.67	-0.33	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00
1.00	2.00	1.00	0.00	1.00	1.00	0.67	-0.33	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67
1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67
2.00	2.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	2.00	0.00
3.00	2.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	2.00	2.00	-1.00
2.00	2.00	1.67	-0.33	2.00	3.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	2.00	2.00	0.00
1.00	2.00	0.67	-0.33	1.00	1.00	1.00	0.00	1.00	1.00	0.67	-0.33	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67
1.00	2.00	2.00	1.00	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	2.67	1.67	1.00	2.00	2.67	1.67
1.00	2.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.67	0.67	1.00	3.00	2.00	1.00	1.00	1.00	2.00	1.00
1.00	1.00	0.67	-0.33	1.00	2.00	1.67	0.67	1.00	1.00	0.67	-0.33	1.00	2.00	2.00	1.00	1.00	1.00	1.67	0.67
3.00	2.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	3.00	1.67	-1.33
3.00	2.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00
1.00	1.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	2.00	2.33	1.33	1.00	1.00	1.67	0.67
2.00	1.00	1.00	-1.00	2.00	2.00	2.00	0.00	2.00	1.00	1.00	-1.00	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33
2.00	1.00	1.67	-0.33	2.00	3.00	2.67	0.67	2.00	3.00	1.67	-0.33	2.00	3.00	3.00	1.00	2.00	2.00	2.00	0.00
3.00	1.00	1.67	-1.33	3.00	3.00	2.33	-0.67	3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00
1.00	2.00	1.67	0.67	1.00	3.00	2.67	1.67	1.00	2.00	1.67	0.67	1.00	2.00	2.33	1.33	1.00	2.00	2.00	1.00
2.00	1.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	2.00	3.00	1.00
1.00	1.00	1.67	0.67	1.00	3.00	1.67	0.67	1.00	1.00	0.67	-0.33	1.00	2.00	2.00	1.00	1.00	1.00	1.67	0.67
2.00	1.00	0.67	-1.33	2.00	3.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	1.00	1.67	-0.33
2.00	1.00	0.67	-1.33	2.00	3.00	1.67	-0.33	2.00	2.00	3.00	1.00	2.00	2.00	2.00	0.00	2.00	1.00	1.67	-0.33
3.00	1.00	0.67	-2.33	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	1.00	1.67	-1.33
3.00	1.00	0.67	-2.33	3.00	3.00	2.67	-0.33	3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00
1.00	1.00	0.67	-0.33	1.00	3.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	3.00	2.67	1.67	1.00	1.00	1.67	0.67
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.00	3.00	2.67	-0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.00	3.00	2.67	-0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00
1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00
3.00	2.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	3.00	0.00	3.00	2.00	2.00	-1.00
3.00	2.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	2.00	1.67	-1.33	3.00	3.00	2.33	-0.67	3.00	2.00	2.00	-1.00
3.00	1.00	0.67	-2.33	3.00	3.00	2.00	-1.00	3.00	1.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	1.00	2.00	-1.00
2.00	1.00	1.00	-1.00	2.00	2.00	1.67	-0.33	2.00	1.00	0.67	-1.33	2.00	0.00	1.67	-0.33	2.00	1.00	1.67	-0.33
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	3.00	0.00
2.00	2.00	1.67	-0.33	2.00	3.00	0.67	-1.33	2.00	1.00	0.67	-1.33	2.00	3.00	1.67	-0.33	2.00	2.00	1.67	-0.33
1.00	1.00	1.67	0.67	1.00	2.00	2.67	1.67	1.00	1.00	2.00	1.00	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00
2.00	2.00	1.67	-0.33	2.00	2.00	2.67	0.67	2.00	1.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	2.00	2.00	0.00
1.00	1.00	1.67	0.67	1.00	3.00	1.67	0.67	1.00	1.00	0.67	-0.33	1.00	2.00	1.67	0.67	1.00	1.00	2.00	1.00
1.00	1.00	0.67	-0.33	1.00	2.00	0.67	-0.33	1.00	1.00	0.67	-0.33	1.00	2.00	1.00	0.00	1.00	1.00	1.67	0.67
73.00	61.00	53.77	-19.23	73.00	96.00	69.75	-3.25	85.00	82.00	69.12	-15.88	79.00	101.00	86.71	7.71	79.00	79.00	83.73	4.73
100.00	83.56	73.66	-26.34	100.00	131.51	95.55	-4.45	100.00	96.47	81.32	-18.68	100.00	127.85	109.76	9.76	100.00	100.00	105.99	5.99

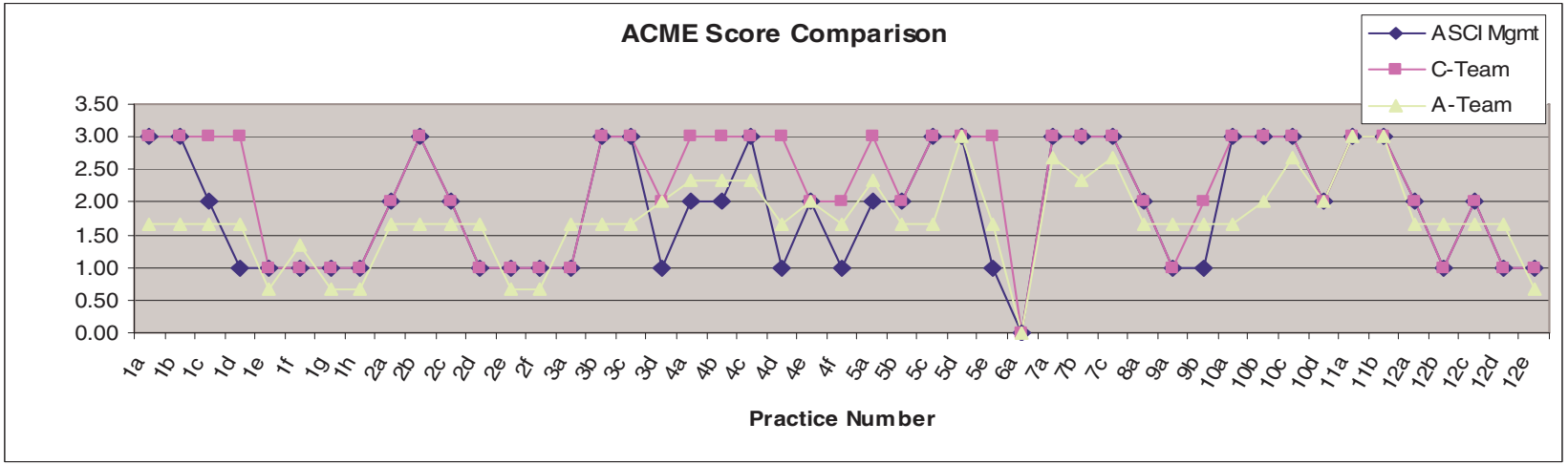
ACME				ITS				CEPTRE				NuGET				SIERRA			
ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP
3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	2.33	-0.67	3.00	3.00	3.00	0.00
3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	2.00	2.33	-0.67	3.00	3.00	3.00	0.00
2.00	3.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	3.00	3.00	1.00
1.00	3.00	1.67	0.67	1.00	3.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	1.00	2.00	1.00	1.00	3.00	1.67	0.67
1.00	1.00	0.67	-0.33	1.00	2.00	1.00	0.00	1.00	1.00	0.67	-0.33	1.00	1.00	2.00	1.00	1.00	1.00	0.67	-0.33
1.00	1.00	1.33	0.33	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67
1.00	1.00	0.67	-0.33	1.00	3.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67
1.00	1.00	0.67	-0.33	1.00	2.00	1.67	0.67	1.00	2.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00
2.00	2.00	1.67	-0.33	2.00	1.00	1.00	-1.00	2.00	3.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33
3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	2.00	1.67	-1.33
2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33
1.00	1.00	1.67	0.67	1.00	2.00	1.00	0.00	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67
1.00	1.00	0.67	-0.33	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00	1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67
1.00	1.00	0.67	-0.33	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	2.00	1.00	1.00	2.00	1.67	0.67
1.00	1.00	1.67	0.67	1.00	3.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67
3.00	3.00	1.67	-1.33	3.00	3.00	2.67	-0.33	3.00	3.00	1.67	-1.33	3.00	2.00	1.67	-1.33	3.00	2.00	1.67	-1.33
3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	2.00	1.67	-1.33
1.00	2.00	2.00	1.00	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	2.00	1.00	1.00	1.00	1.67	0.67
2.00	3.00	2.33	0.33	2.00	3.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	1.00	1.00	-1.00
2.00	3.00	2.33	0.33	2.00	3.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33
3.00	3.00	2.33	-0.67	3.00	3.00	2.33	-0.67	3.00	3.00	1.67	-1.33	3.00	2.00	2.00	-1.00	3.00	2.00	1.67	-1.33
1.00	3.00	1.67	0.67	1.00	3.00	2.00	1.00	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	2.00	1.00
2.00	2.00	2.00	0.00	2.00	3.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33
1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	0.67	-0.33	1.00	1.00	2.67	1.67	1.00	3.00	2.00	1.00
2.00	3.00	2.33	0.33	2.00	3.00	1.67	-0.33	2.00	2.00	1.00	-1.00	2.00	3.00	2.67	0.67	2.00	3.00	3.00	1.00
2.00	2.00	1.67	-0.33	2.00	3.00	2.33	0.33	2.00	2.00	1.00	-1.00	2.00	3.00	2.67	0.67	2.00	3.00	3.00	1.00
3.00	3.00	1.67	-1.33	3.00	2.00	2.00	-1.00	3.00	3.00	1.00	-2.00	3.00	3.00	2.00	-1.00	3.00	3.00	3.00	0.00
3.00	3.00	3.00	0.00	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	2.00	2.00	-1.00	3.00	3.00	3.00	0.00
1.00	3.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3.00	3.00	2.67	-0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3.00	3.00	2.33	-0.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3.00	3.00	2.67	-0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2.00	2.00	1.67	-0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.00	1.00	1.67	0.67	1.00	3.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.67	0.67	1.00	2.00	2.67	1.67
1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	3.00	2.67	1.67
3.00	3.00	1.67	-1.33	3.00	2.00	1.00	-2.00	3.00	3.00	1.00	-2.00	3.00	2.00	2.67	-0.33	3.00	2.00	1.67	-1.33
3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	2.00	2.00	-1.00	3.00	2.00	2.33	-0.67	3.00	3.00	1.67	-1.33
3.00	3.00	2.67	-0.33	3.00	3.00	2.00	-1.00	3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	2.00	1.67	-1.33
2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	1.00	0.67	-1.33	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33
3.00	3.00	3.00	0.00	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	2.00	2.00	-1.00
3.00	3.00	3.00	0.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	3.00	0.00	3.00	2.00	2.00	-1.00
2.00	2.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	2.00	1.00	-1.00	2.00	1.00	0.67	-1.33
1.00	1.00	1.67	0.67	1.00	3.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67
2.00	2.00	1.67	-0.33	2.00	3.00	1.67	-0.33	2.00	2.00	1.00	-1.00	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33
1.00	1.00	1.67	0.67	1.00	3.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67
1.00	1.00	0.67	-0.33	1.00	3.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67
90.00	103.00	82.09	-7.91	79.00	108.00	72.07	-6.93	79.00	86.00	62.75	-16.25	79.00	82.00	83.06	4.06	79.00	90.00	80.76	1.76
100.00	114.44	91.21	-8.79	100.00	136.71	91.23	-8.77	100.00	108.86	79.43	-20.57	100.00	103.80	105.14	5.14	100.00	113.92	102.23	2.23

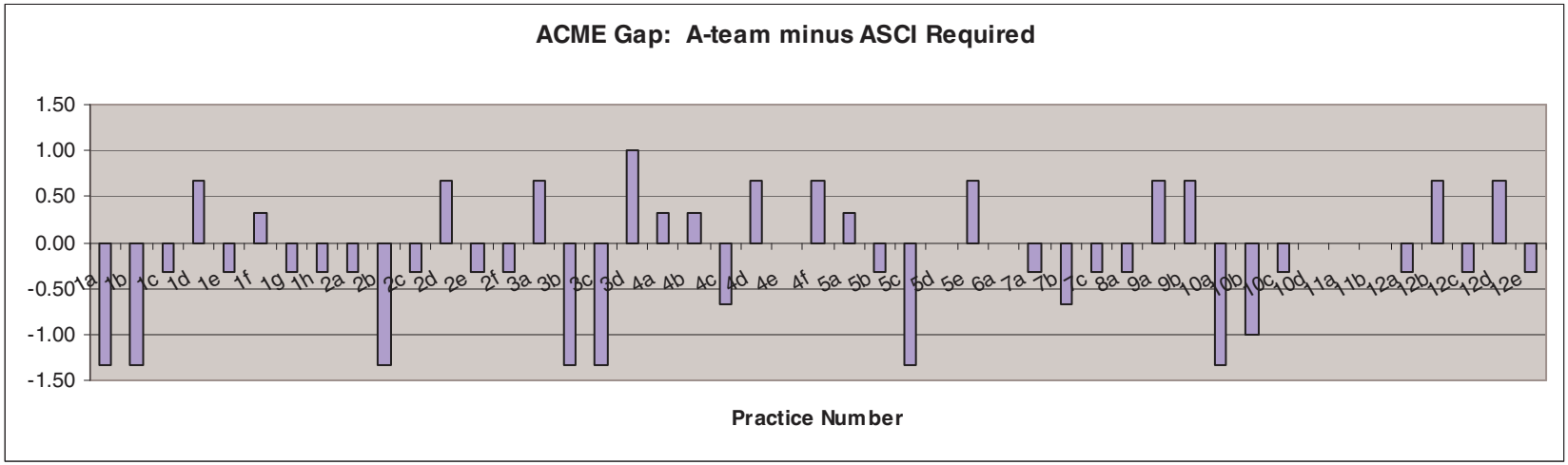
Zoltan				ALEGRA				NEVADA				EMPHASIS/CABANA				CUBIT/Verde			
ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP	ASCI Mgmt	C-Team	A-Team	GAP
3.00	3.00	2.67	-0.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.33	-0.67	3.00	3.00	1.67	-1.33
3.00	3.00	2.67	-0.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	2.00	1.67	-1.33	3.00	3.00	1.67	-1.33
2.00	3.00	2.67	0.67	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	2.00	2.33	0.33	2.00	2.00	1.67	-0.33
1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	1.00	2.00	1.00	1.00	2.00	1.67	0.67
1.00	2.00	0.67	-0.33	1.00	2.00	1.00	0.00	1.00	2.00	1.00	0.00	1.00	2.00	2.00	1.00	1.00	1.00	1.67	0.67
1.00	2.00	1.67	0.67	1.00	1.00	0.67	-0.33	1.00	1.00	0.67	-0.33	1.00	1.00	1.00	0.00	1.00	1.00	0.67	-0.33
1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	0.67	-0.33
1.00	3.00	3.00	2.00	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67
2.00	2.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33
3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.67	-0.33	3.00	3.00	1.67	-1.33
2.00	3.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	3.00	3.00	1.00	2.00	3.00	1.67	-0.33
1.00	1.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.67	0.67
1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	1.00	1.67	0.67
1.00	3.00	2.33	1.33	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	3.00	2.00	1.00	1.00	1.00	1.67	0.67
1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67
3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	1.67	-1.33	3.00	3.00	2.33	-0.67	3.00	3.00	2.00	-1.00
3.00	3.00	1.67	-1.33	3.00	3.00	2.00	-1.00	3.00	3.00	2.00	-1.00	3.00	3.00	2.33	-0.67	3.00	3.00	1.67	-1.33
1.00	3.00	2.33	1.33	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00	1.00	1.00	1.67	0.67
2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	1.67	-0.33	2.00	2.00	2.00	0.00
2.00	3.00	1.67	-0.33	2.00	3.00	2.00	0.00	2.00	3.00	2.00	0.00	2.00	2.00	2.33	0.33	2.00	3.00	2.00	0.00
3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00	3.00	2.00	2.00	-1.00	3.00	3.00	2.67	-0.33
1.00	3.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00	1.00	3.00	2.67	1.67
2.00	2.00	1.67	-0.33	2.00	3.00	3.00	1.00	2.00	3.00	3.00	1.00	2.00	2.00	1.67	-0.33	2.00	3.00	2.67	0.67
1.00	3.00	2.33	1.33	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67
2.00	2.00	2.67	0.67	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	3.00	1.67	-0.33
2.00	2.00	2.67	0.67	2.00	3.00	3.00	1.00	2.00	3.00	3.00	1.00	2.00	2.00	1.67	-0.33	2.00	3.00	3.00	1.00
3.00	3.00	2.67	-0.33	3.00	3.00	2.33	-0.67	3.00	3.00	2.33	-0.67	3.00	2.00	1.67	-1.33	3.00	3.00	2.33	-0.67
3.00	3.00	2.67	-0.33	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00	3.00	2.00	1.00	-2.00	3.00	3.00	2.00	-1.00
1.00	3.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	2.00	1.67	0.67
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	3.00	3.00	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	3.00	3.00	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.00	2.00	2.00	1.00	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67
1.00	2.00	2.33	1.33	1.00	1.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00
3.00	2.00	1.67	-1.33	3.00	2.00	2.00	-1.00	3.00	2.00	2.00	-1.00	3.00	2.00	2.00	-1.00	3.00	3.00	2.00	-1.00
3.00	3.00	2.00	-1.00	3.00	2.00	2.00	-1.00	3.00	2.00	2.00	-1.00	3.00	2.00	2.00	-1.00	3.00	3.00	1.67	-1.33
3.00	3.00	2.67	-0.33	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00	3.00	1.00	1.00	-2.00	3.00	3.00	1.67	-1.33
2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	1.00	1.67	-0.33	2.00	1.00	1.67	-0.33
3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00	3.00	2.00	2.00	-1.00	3.00	2.00	1.67	-1.33
3.00	3.00	1.67	-1.33	3.00	3.00	3.00	0.00	3.00	3.00	3.00	0.00	3.00	2.00	2.00	-1.00	3.00	2.00	1.67	-1.33
2.00	2.00	1.67	-0.33	2.00	1.00	1.67	-0.33	2.00	1.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	2.00	1.67	-0.33
1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00
2.00	2.00	1.67	-0.33	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00	2.00	2.00	2.00	0.00
1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00	1.00	2.00	2.00	1.00	1.00	2.00	1.67	0.67	1.00	2.00	2.00	1.00
1.00	2.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	1.00	1.67	0.67	1.00	1.00	1.00	0.00	1.00	2.00	2.00	1.00
79.00	103.00	82.10	3.10	85.00	93.00	85.37	0.37	79.00	87.00	79.04	0.04	79.00	77.00	73.69	-5.31	79.00	92.00	76.43	-2.57
100.00	130.38	103.92	3.92	100.00	109.41	100.44	0.44	100.00	110.13	100.05	0.05	100.00	97.47	93.28	-6.72	100.00	116.46	96.75	-3.25

Appendix B Individual Code Team Results

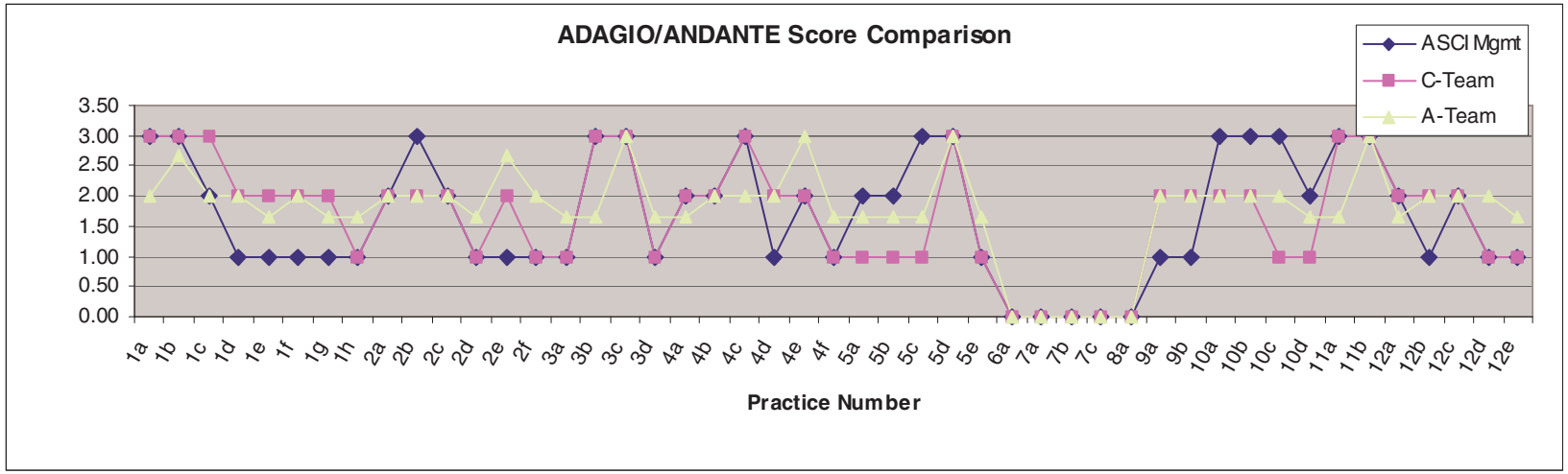
(listed in alphabetical order)

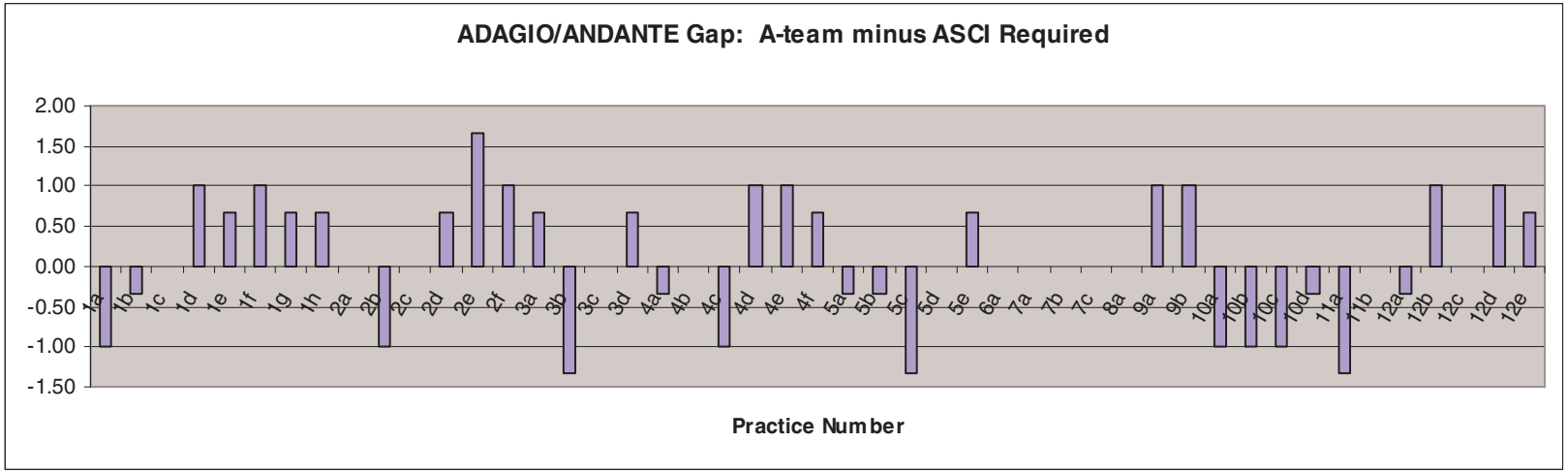
ACME



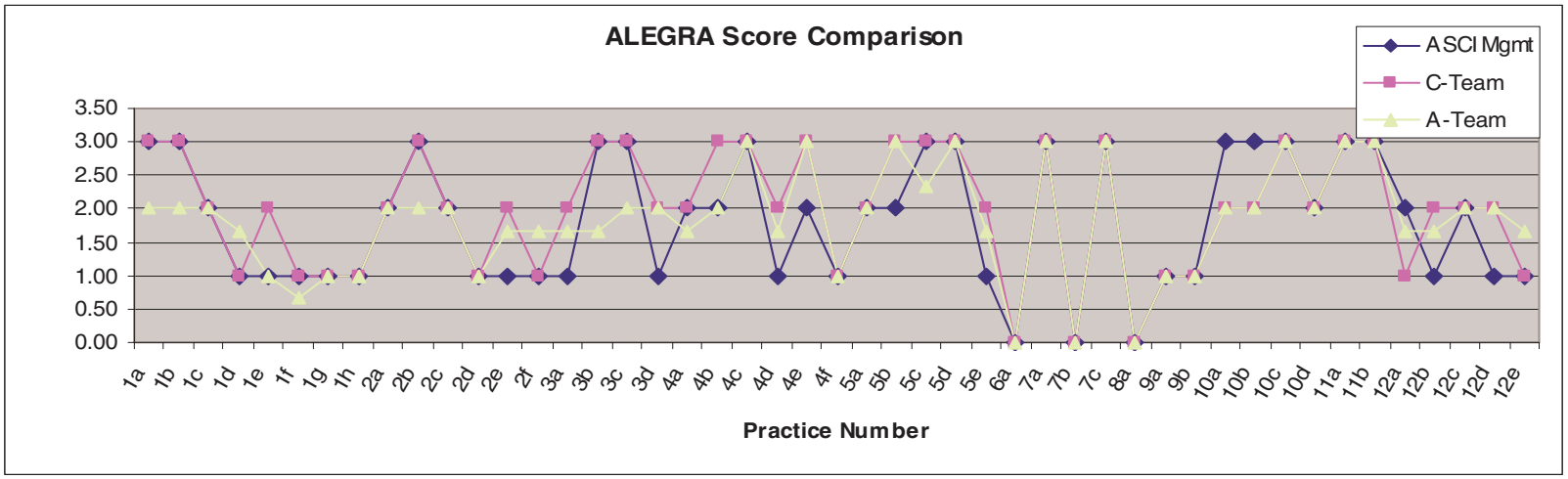


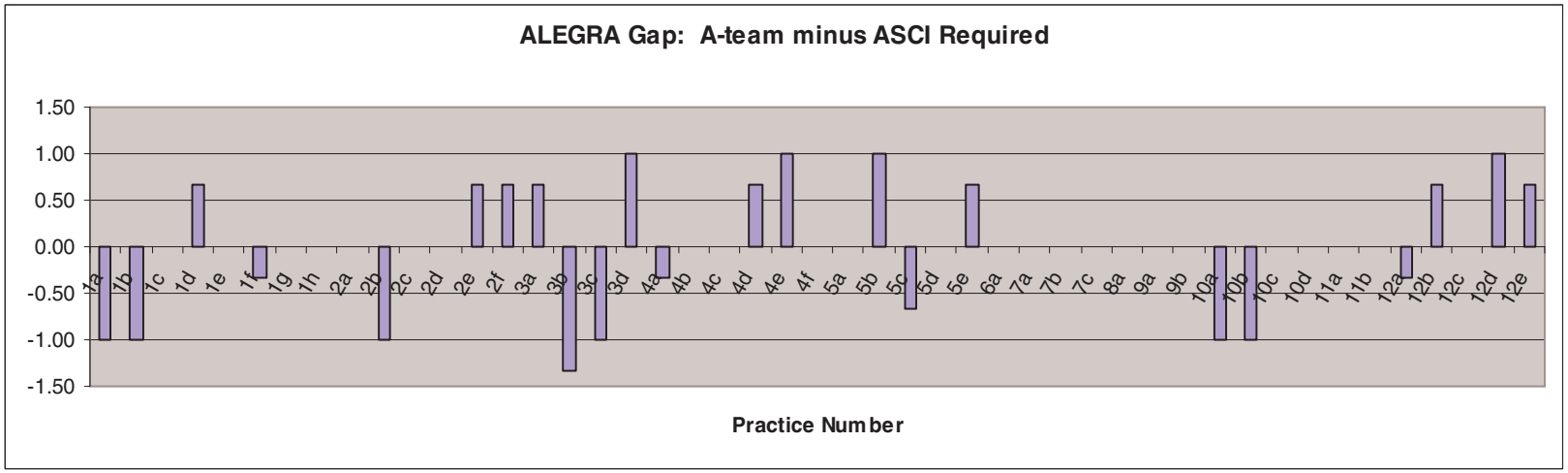
ADAGIO/ANDANTE



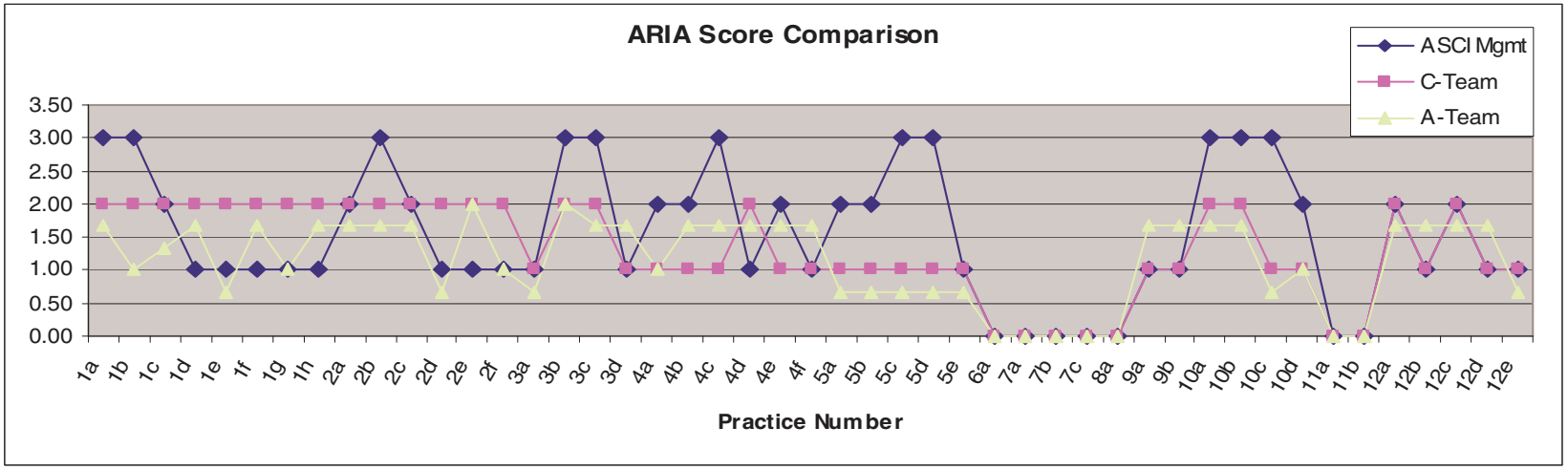


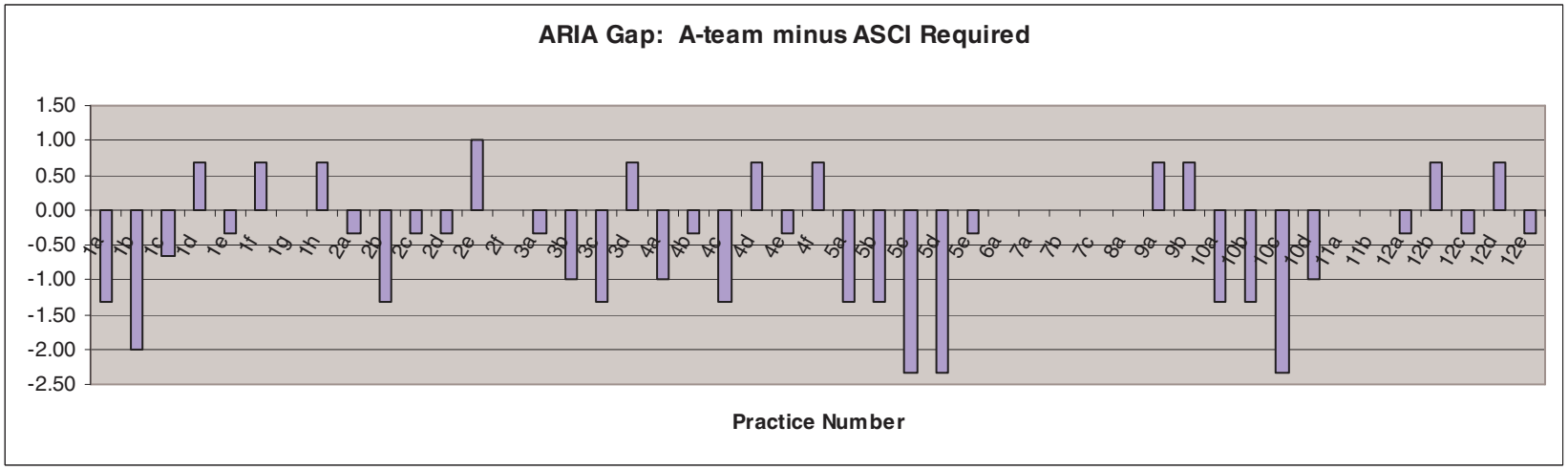
ALEGRA


ALEGRA Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

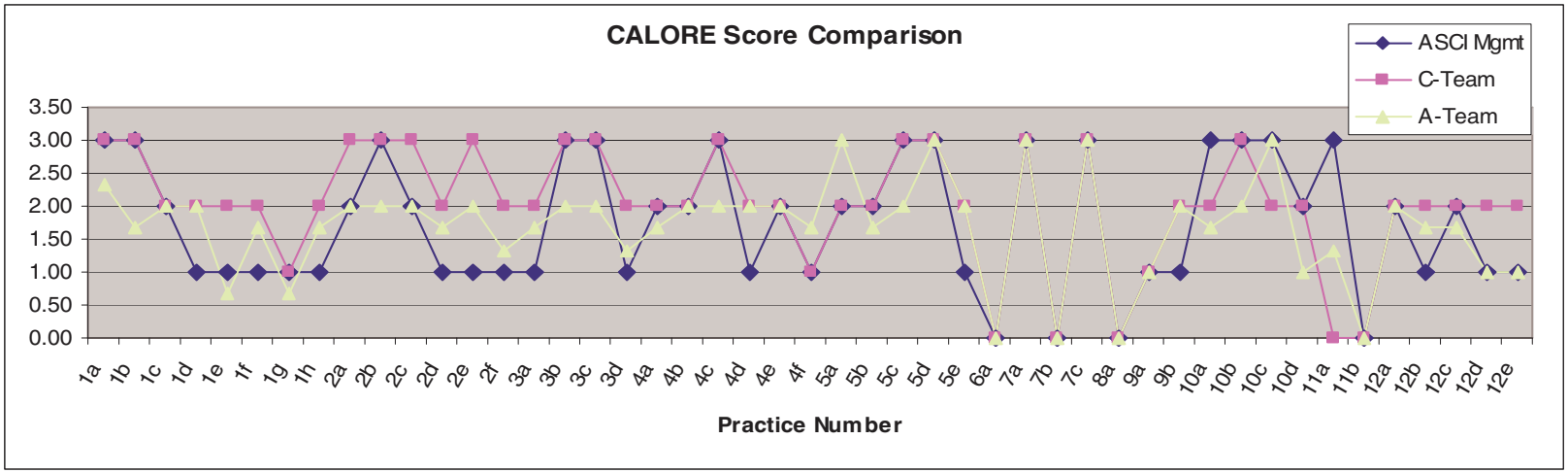

ALEGRA Gap: A-team minus ASCI Required
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

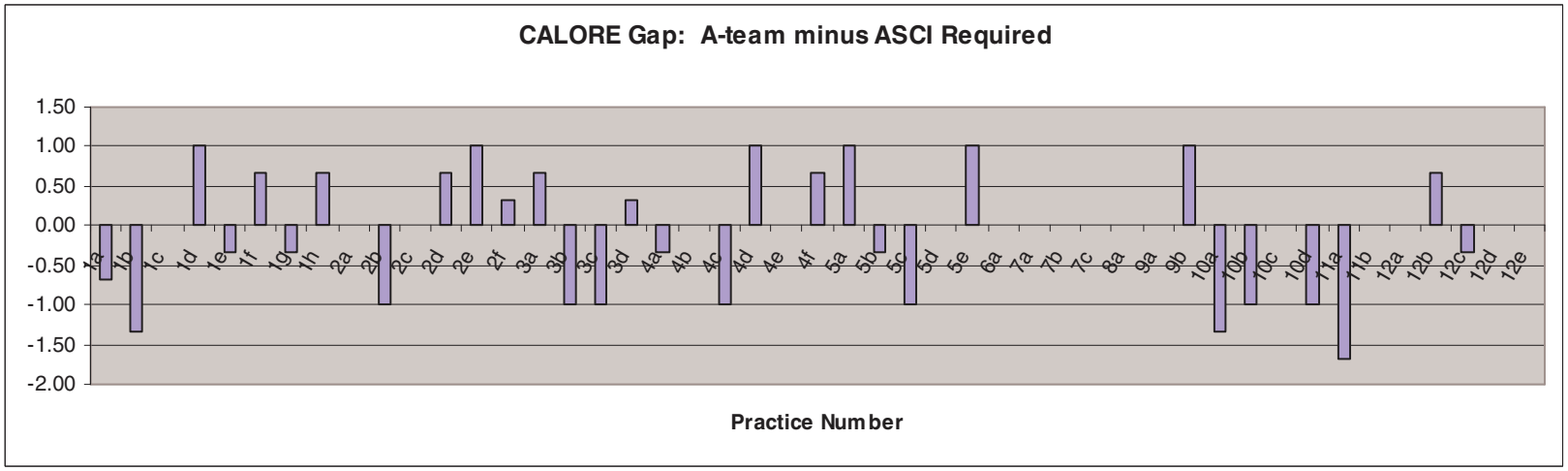
ARIA


ARIA Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

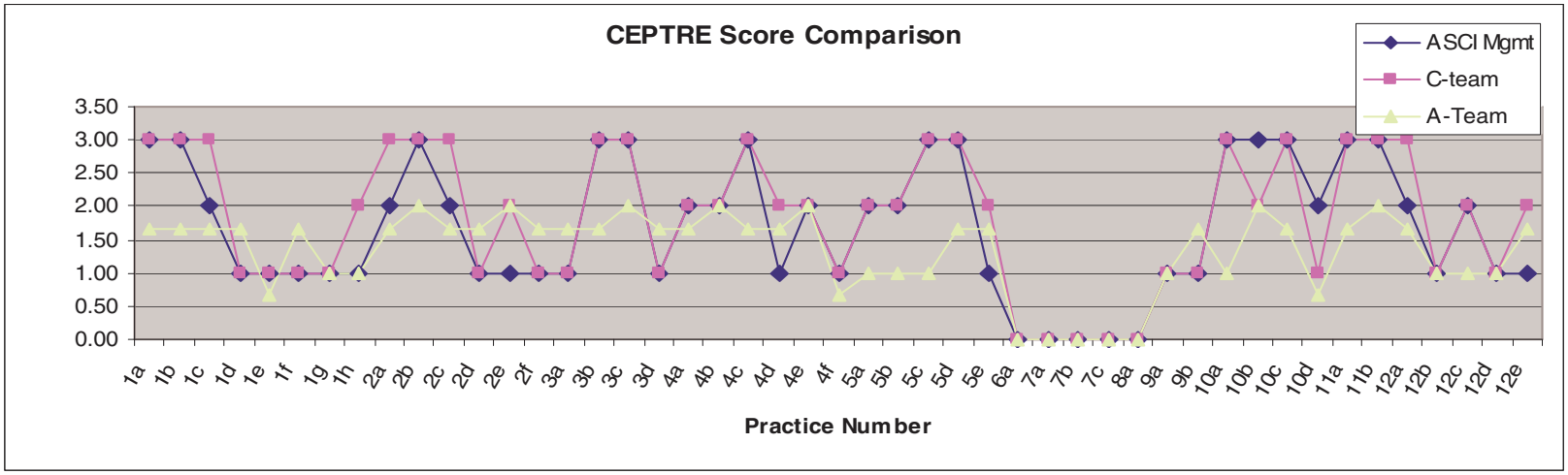

ARIA Gap: A-team minus ASCI Required
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
-2.00
-2.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

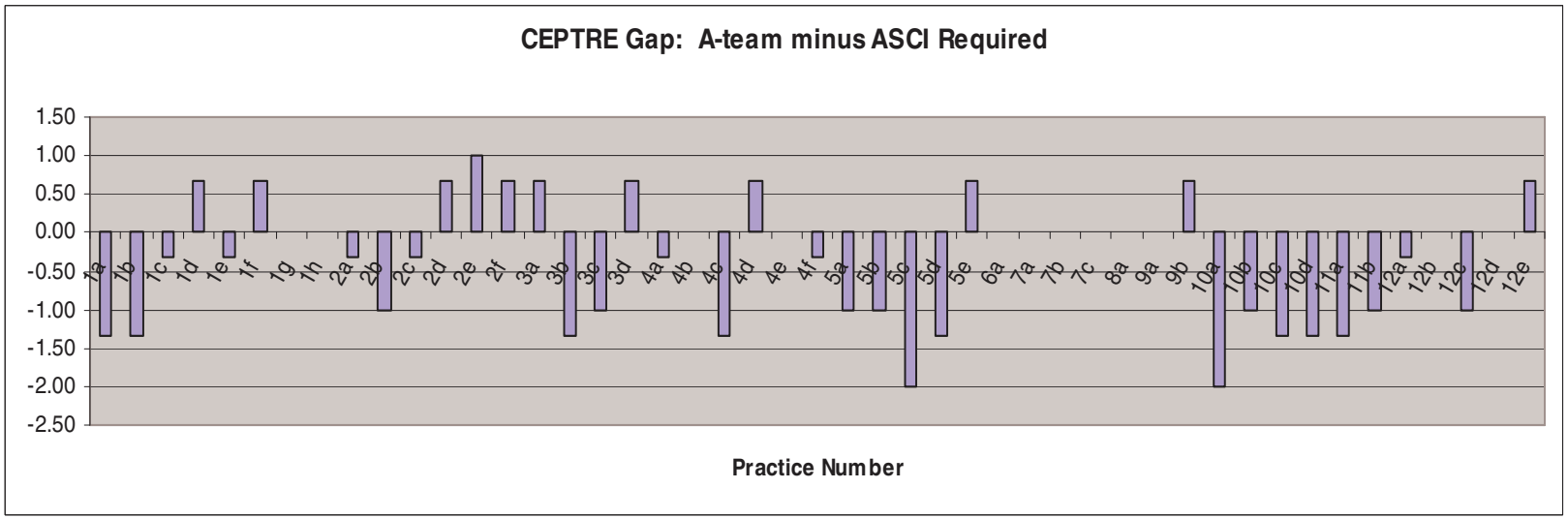
CALORE


CALORE Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

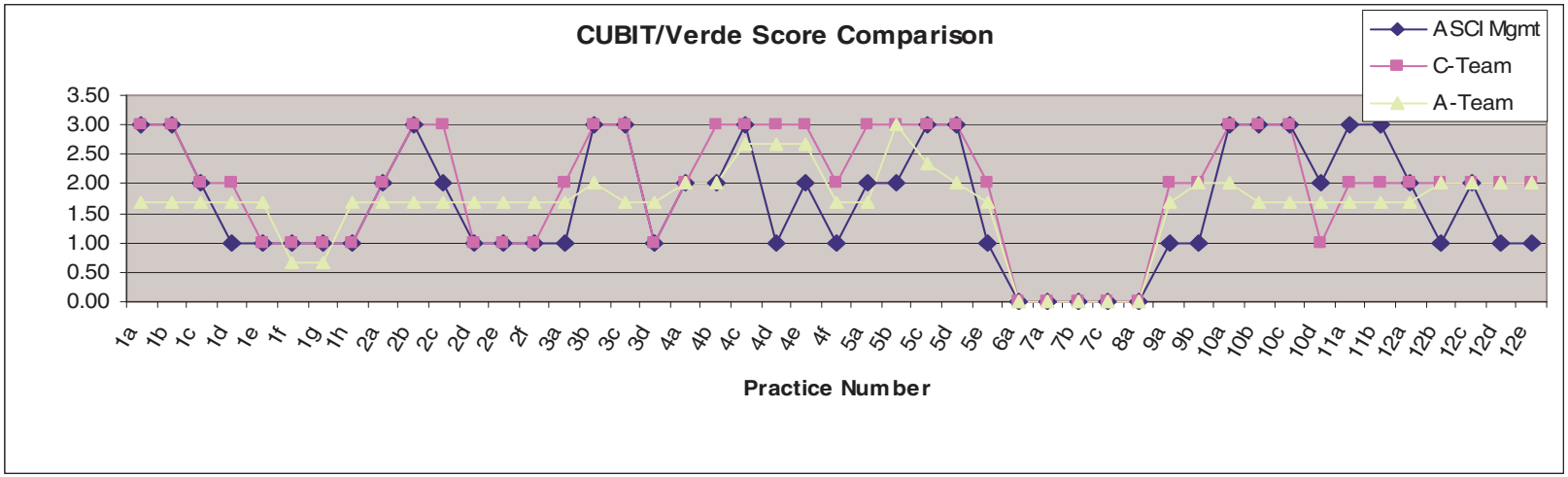

CALORE Gap: A-team minus ASCI Required
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
-2.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

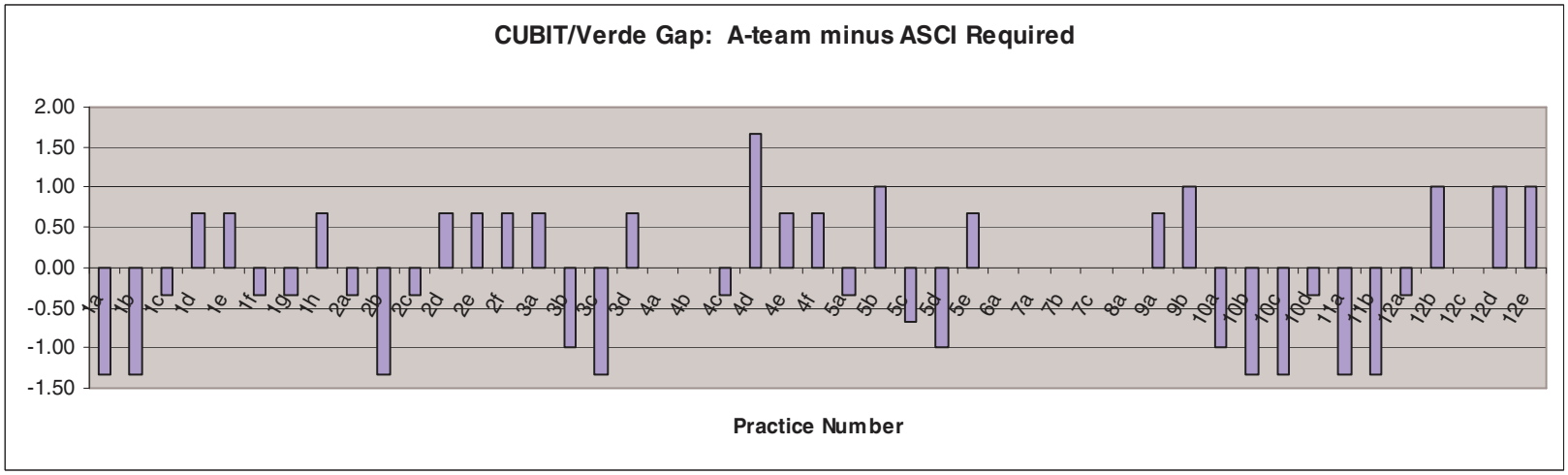
CEPTRE


CEPTRE Score Comparison
ASCI Mgmt
C-team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

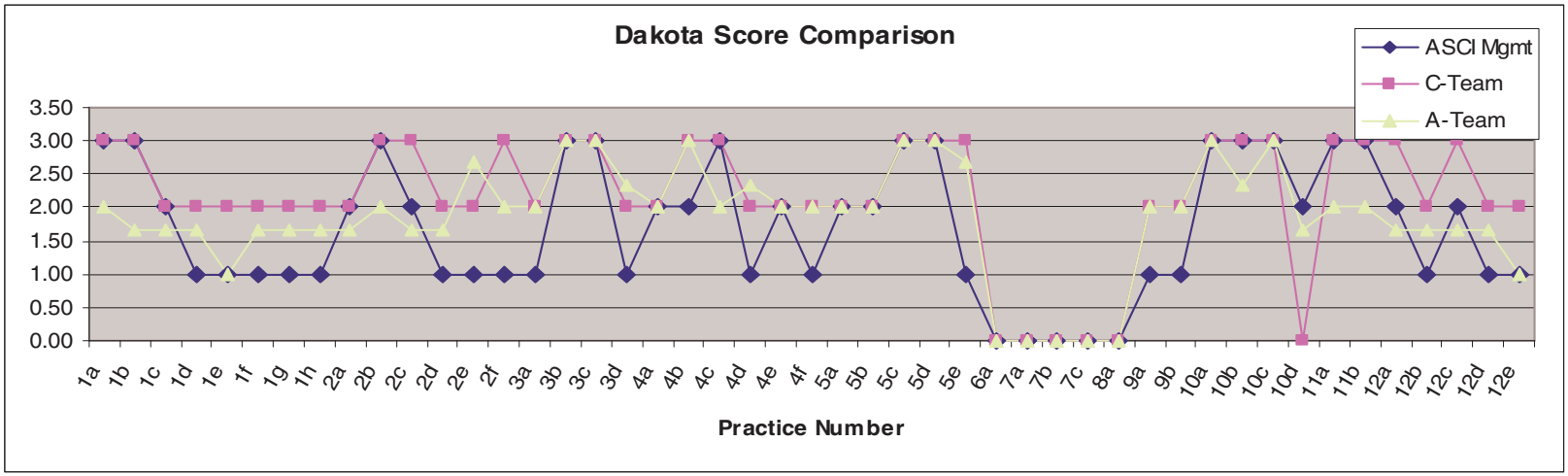

CEPTRE Gap: A-team minus ASCI Required
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
-2.00
-2.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

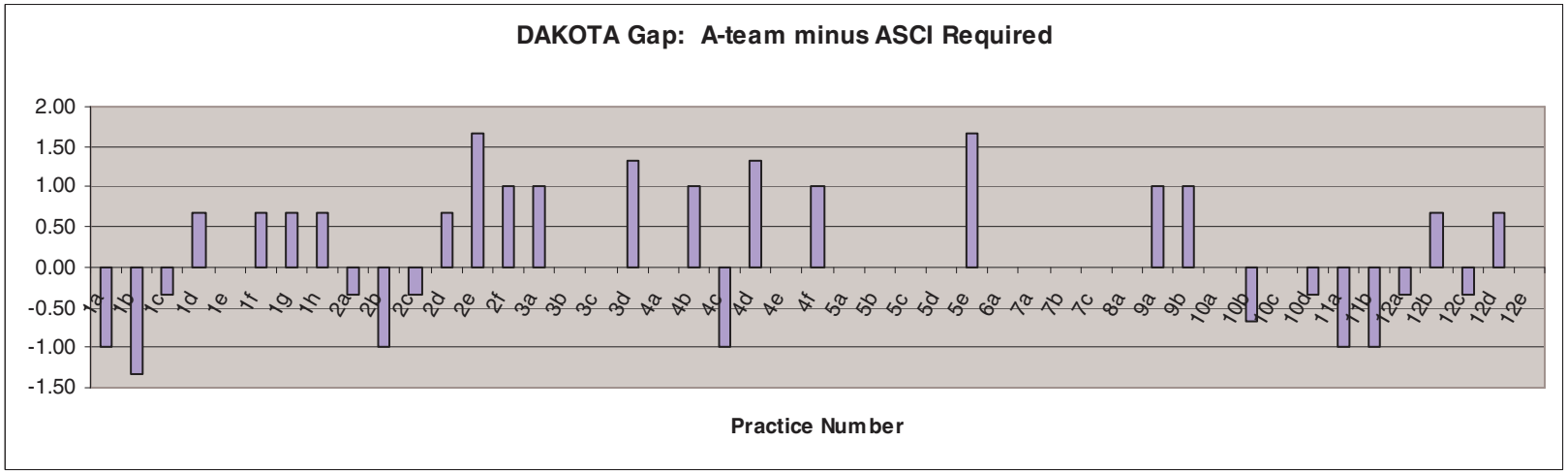
CUBIT/Verde


CUBIT/Verde Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

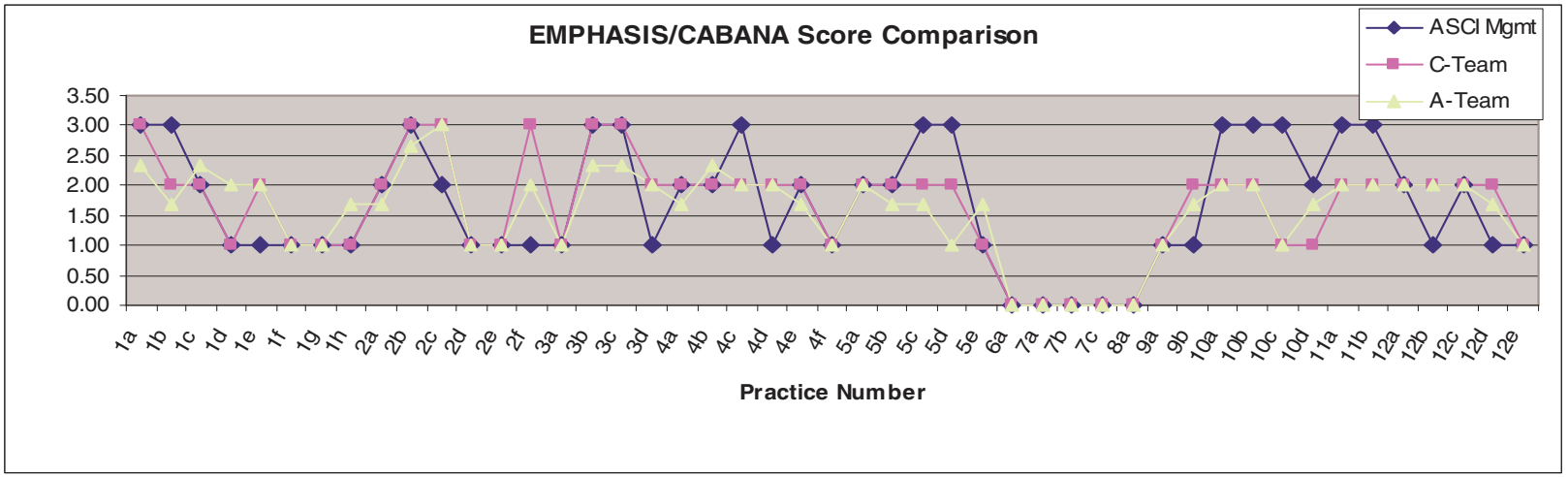

CUBIT/Verde Gap: A-team minus ASCI Required
2.00
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

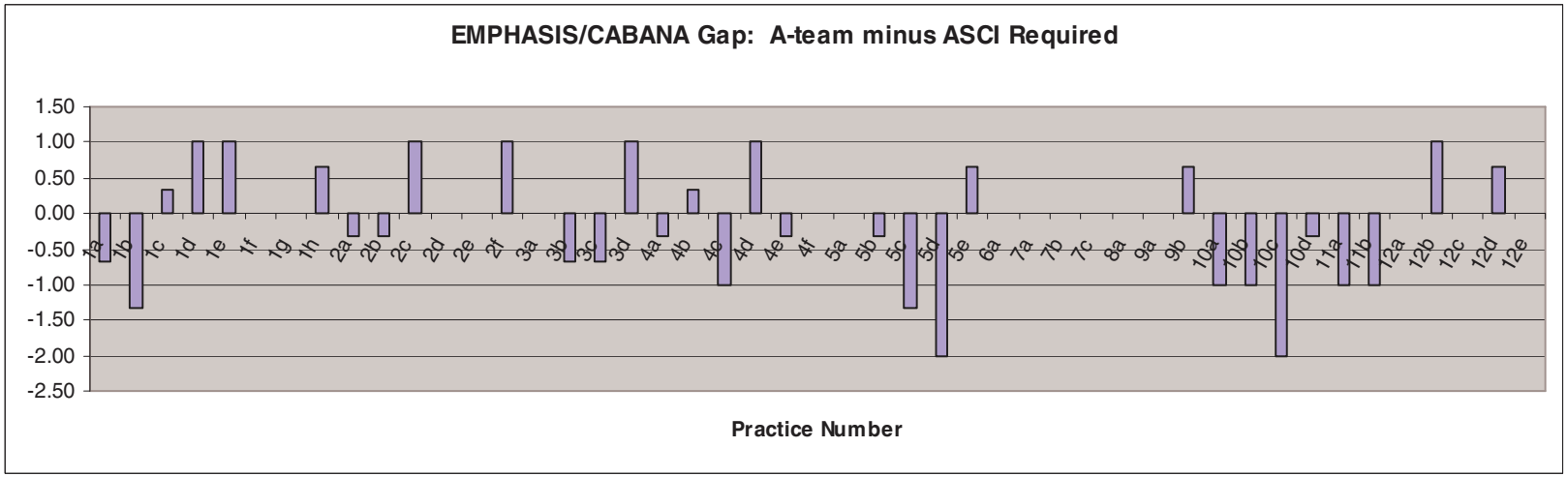
Dakota


Dakota Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

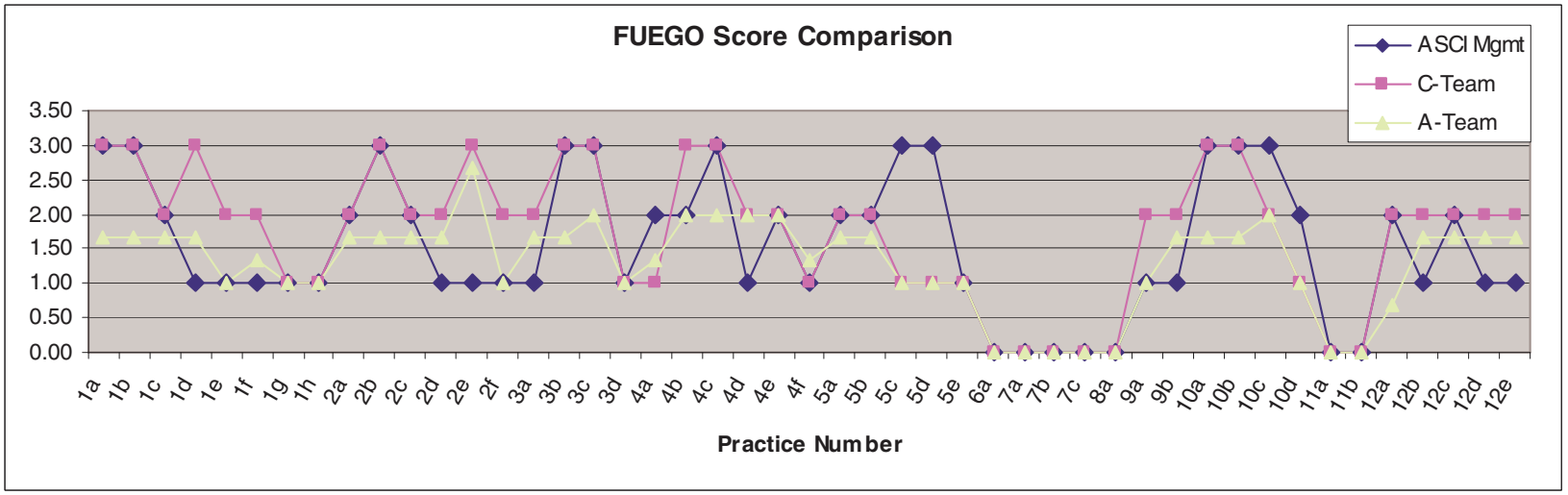

DAKOTA Gap: A-team minus ASCI Required
2.00
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

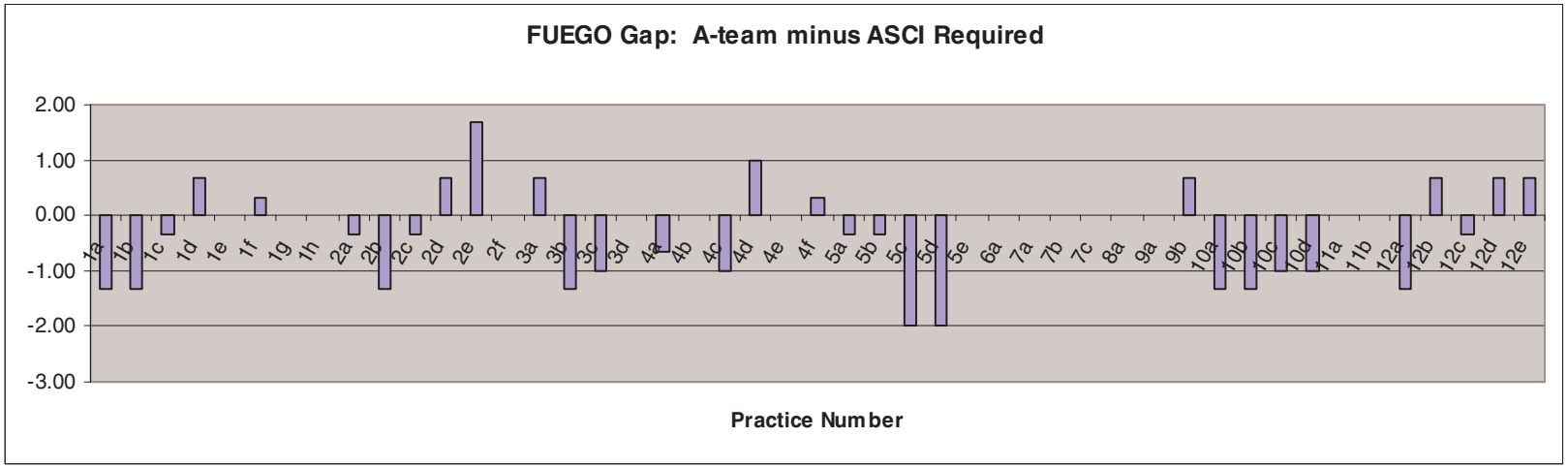
EMPHASIS/CABANA



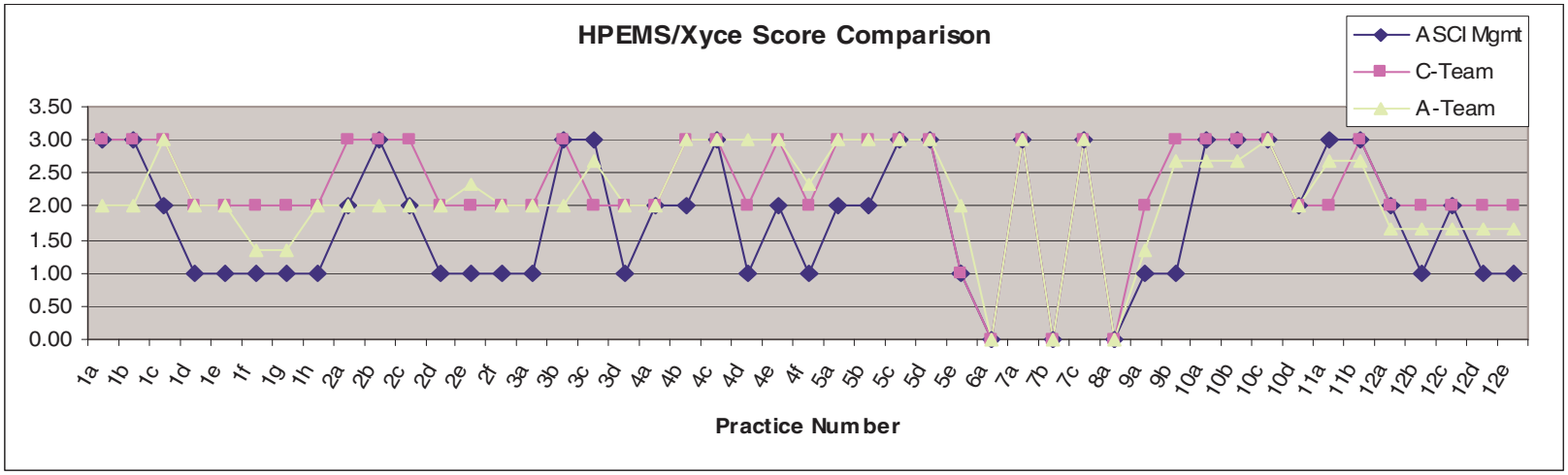


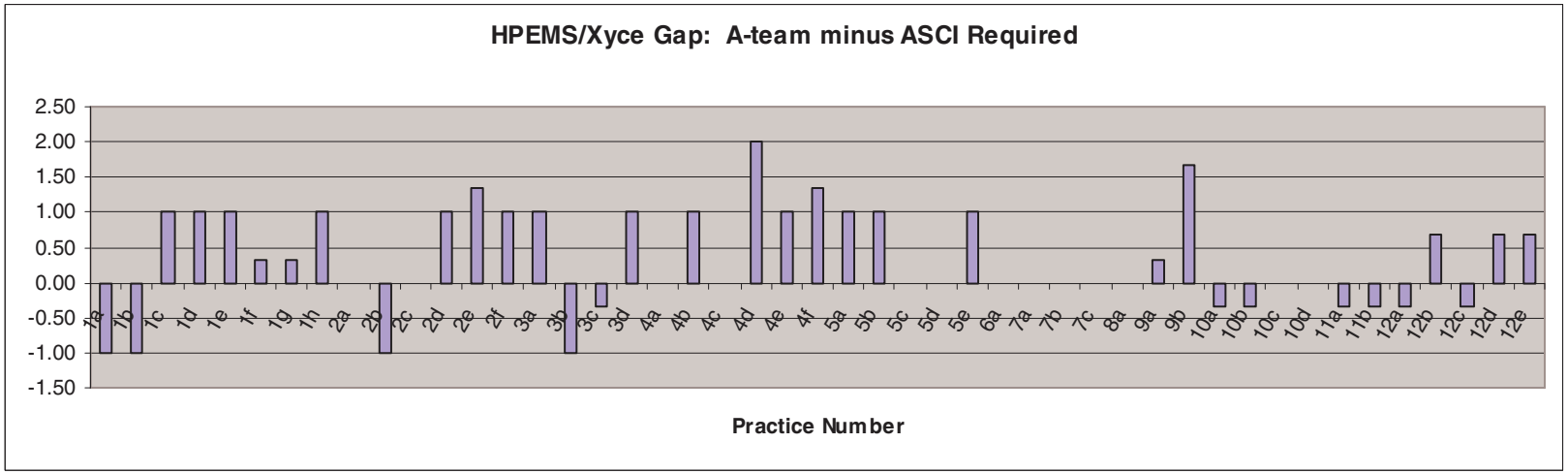
FUEGO


FUEGO Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

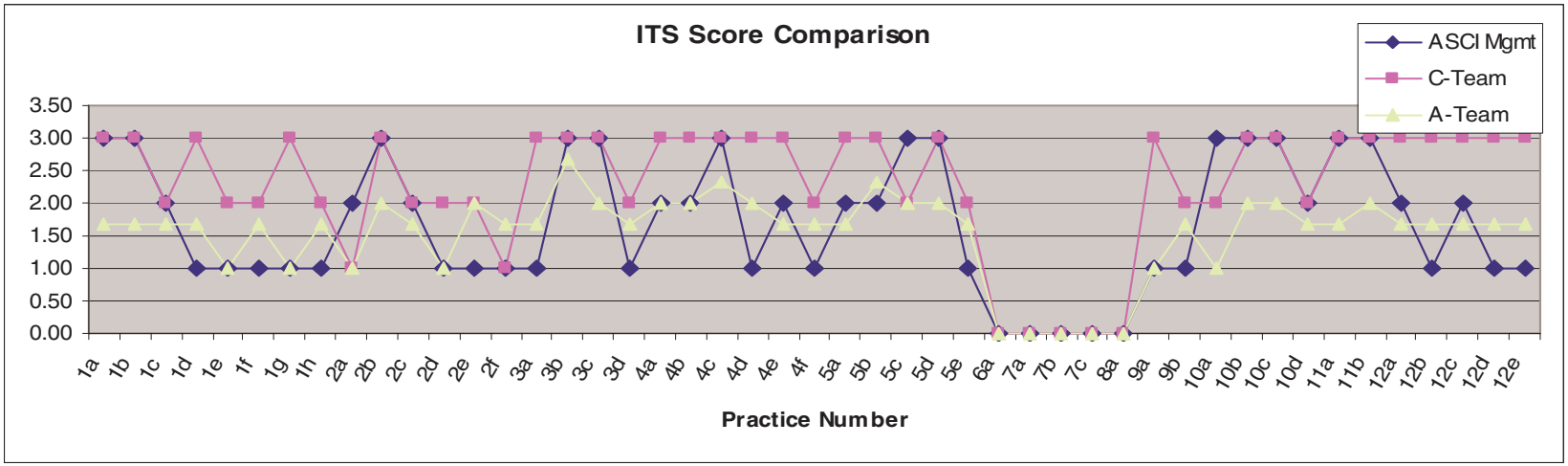

FUEGO Gap: A-team minus ASCI Required
2.00
1.00
0.00
-1.00
-2.00
-3.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

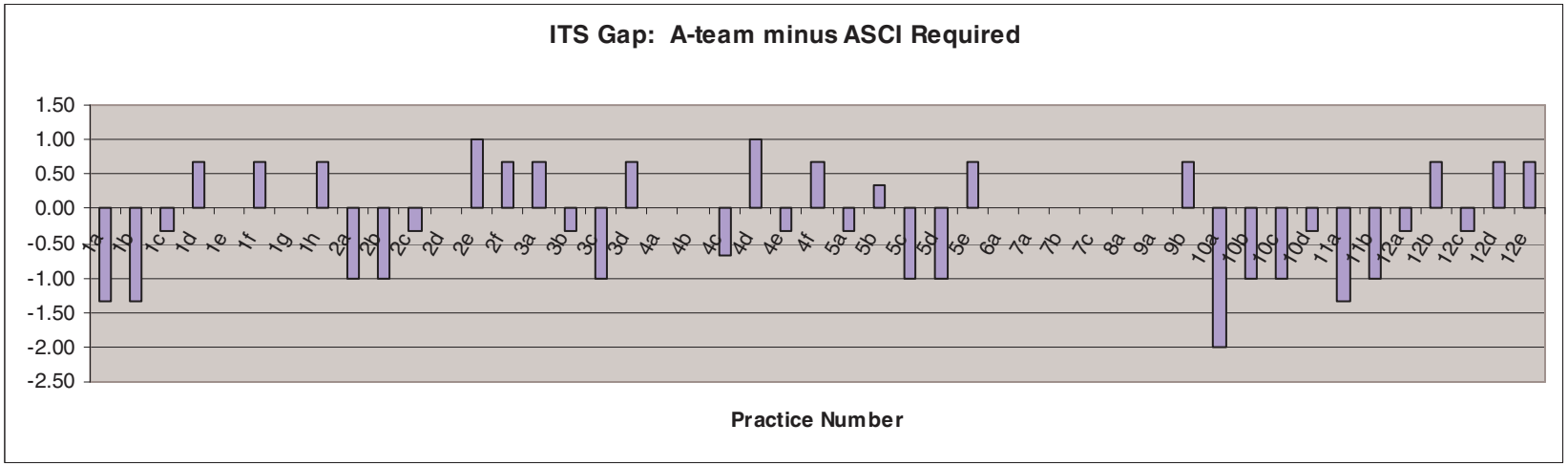
HPEMS/Xyce


HPEMS/Xyce Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

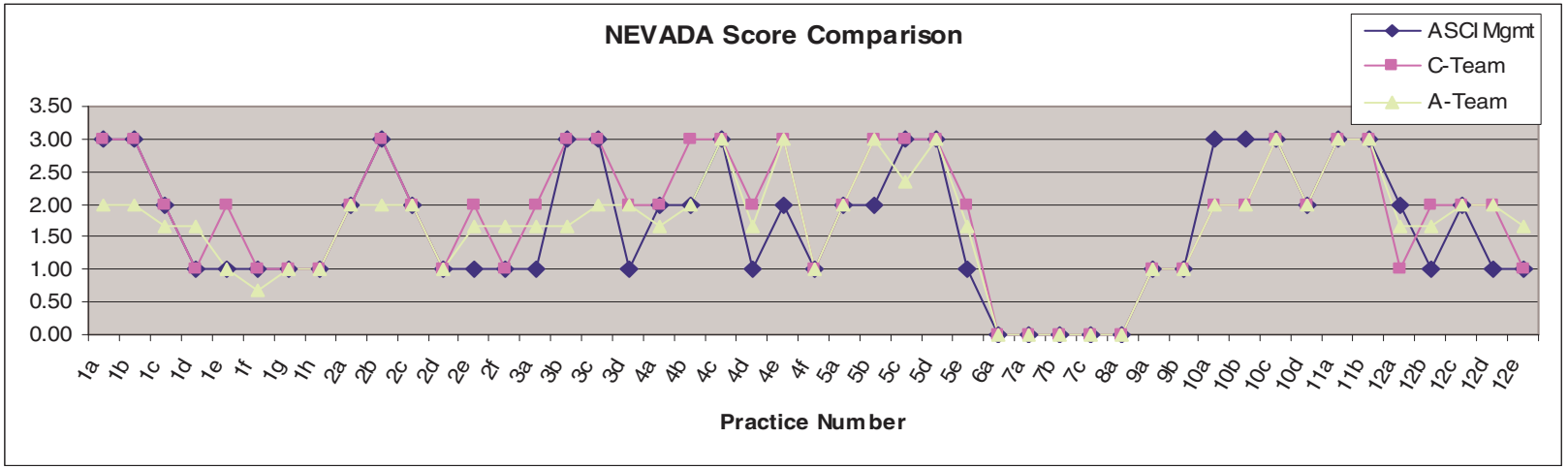

HPEMS/Xyce Gap: A-team minus ASCI Required
2.50
2.00
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

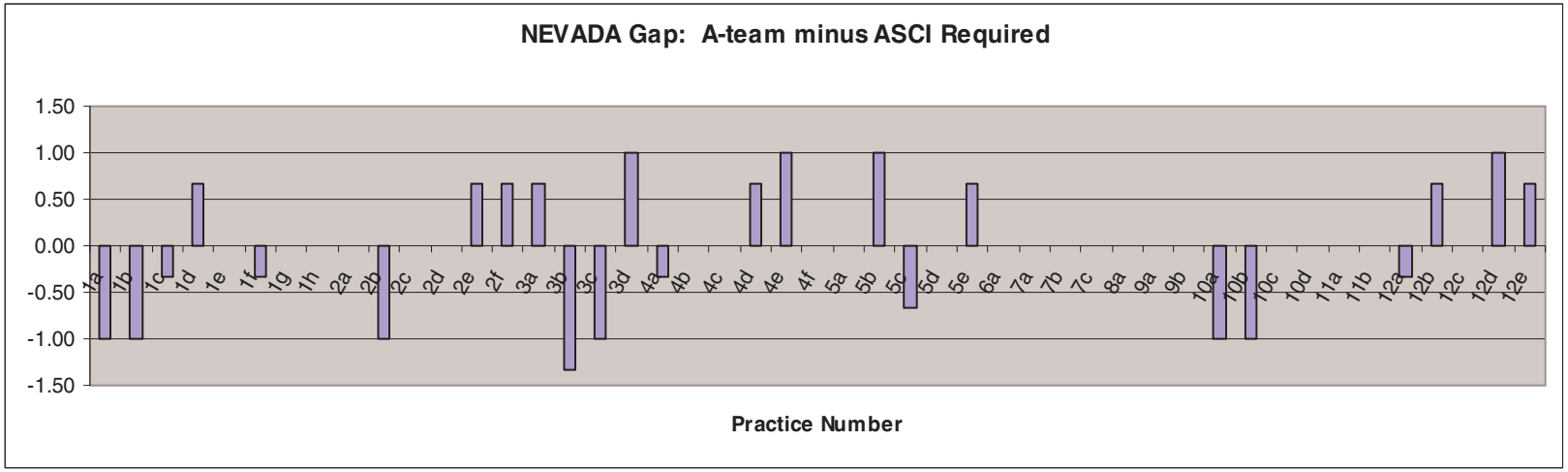
ITS


ITS Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

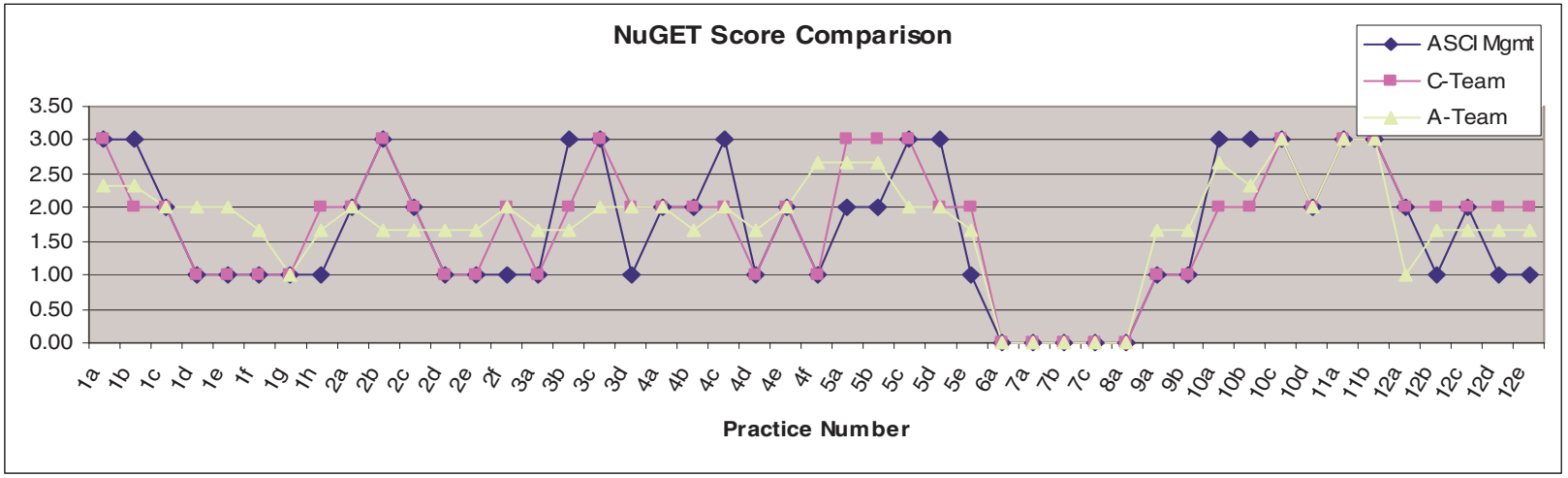

ITS Gap: A-team minus ASCI Required
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
-2.00
-2.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

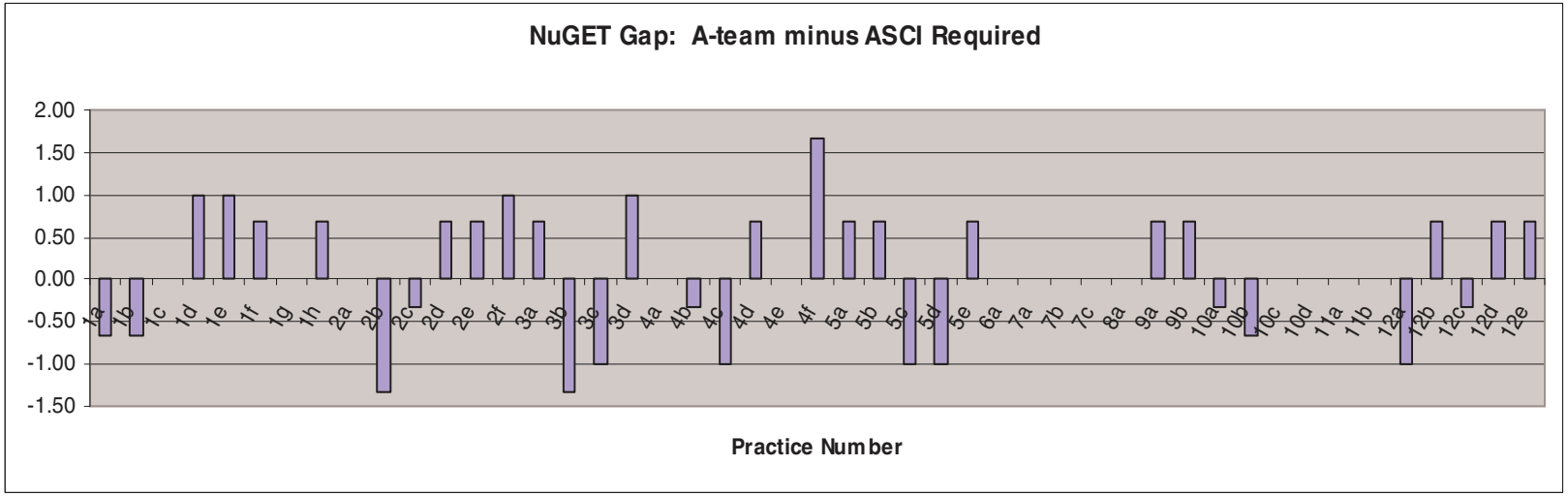
NEVADA


NEVADA Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

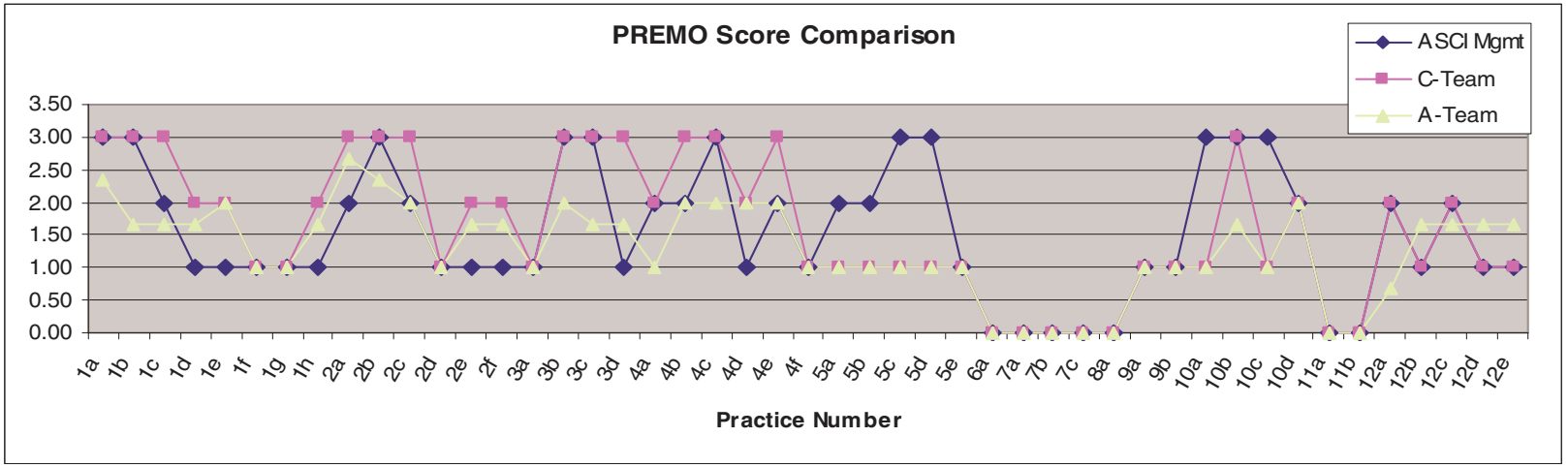

NEVADA Gap: A-team minus ASCI Required
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

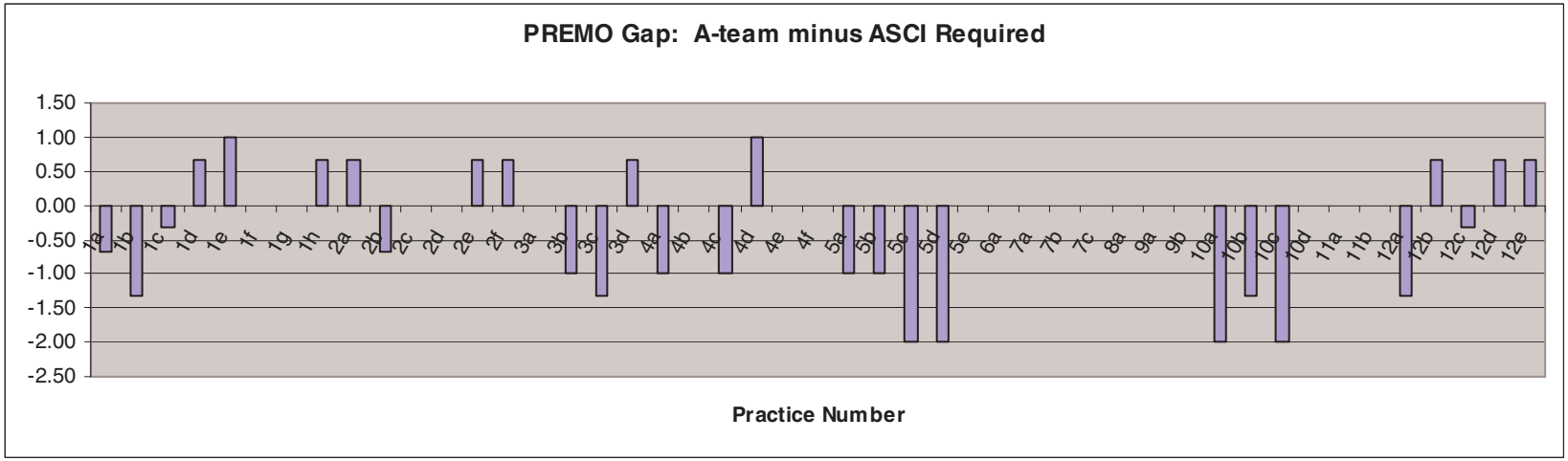
NuGET


NuGET Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

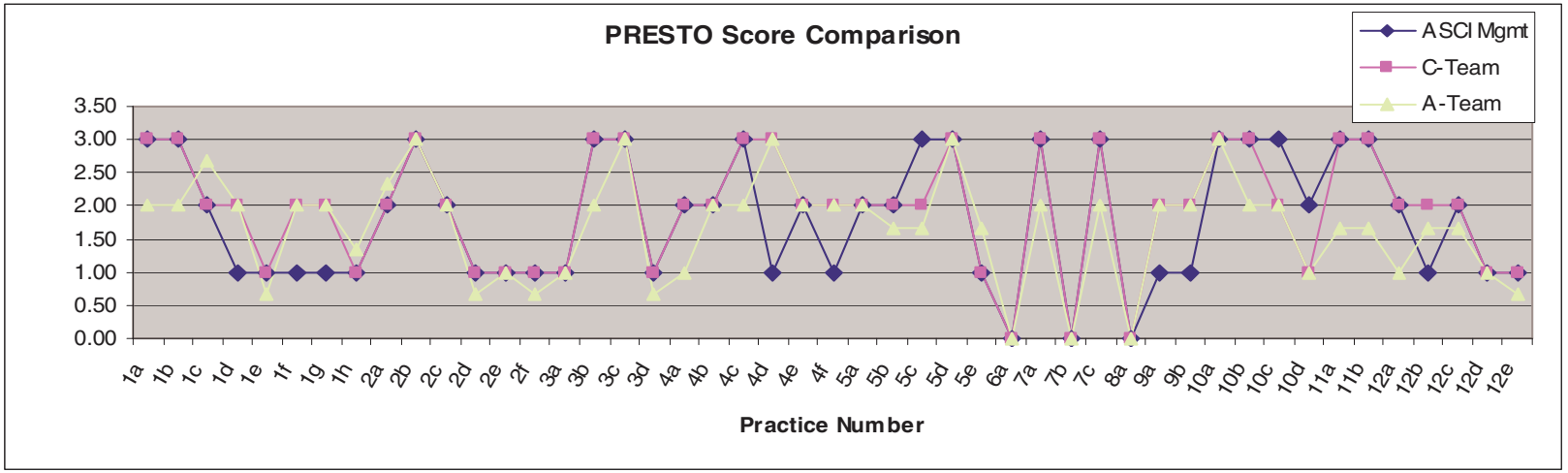

NuGET Gap: A-team minus ASCI Required
2.00
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

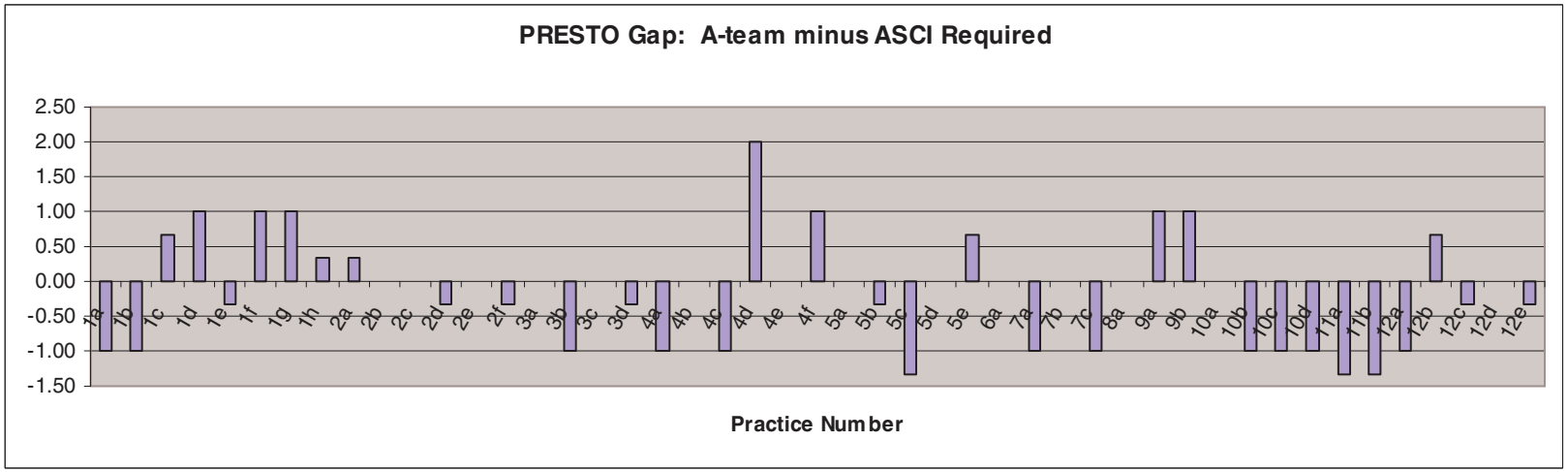
PREMO


PREMO Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

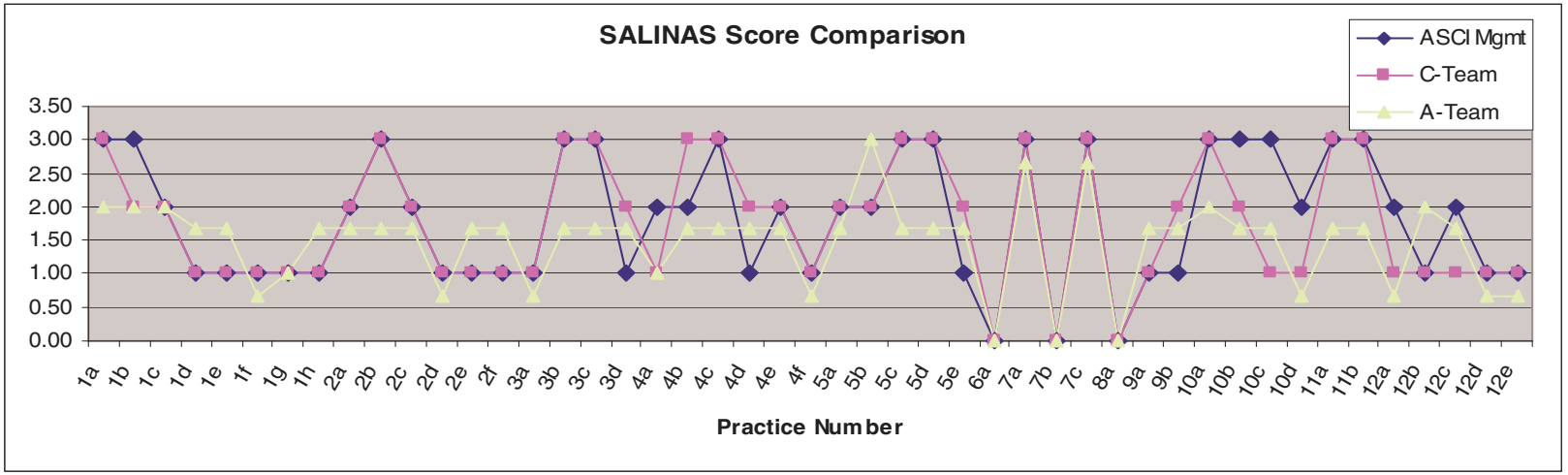

PREMO Gap: A-team minus ASCI Required
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
-2.00
-2.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

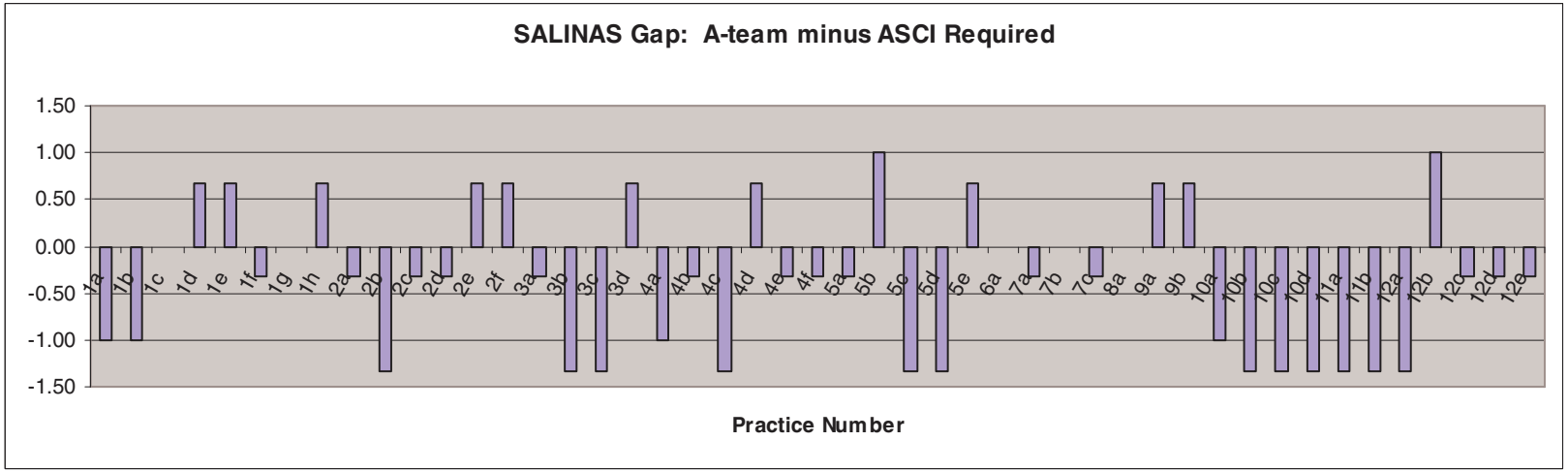
PRESTO


PRESTO Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

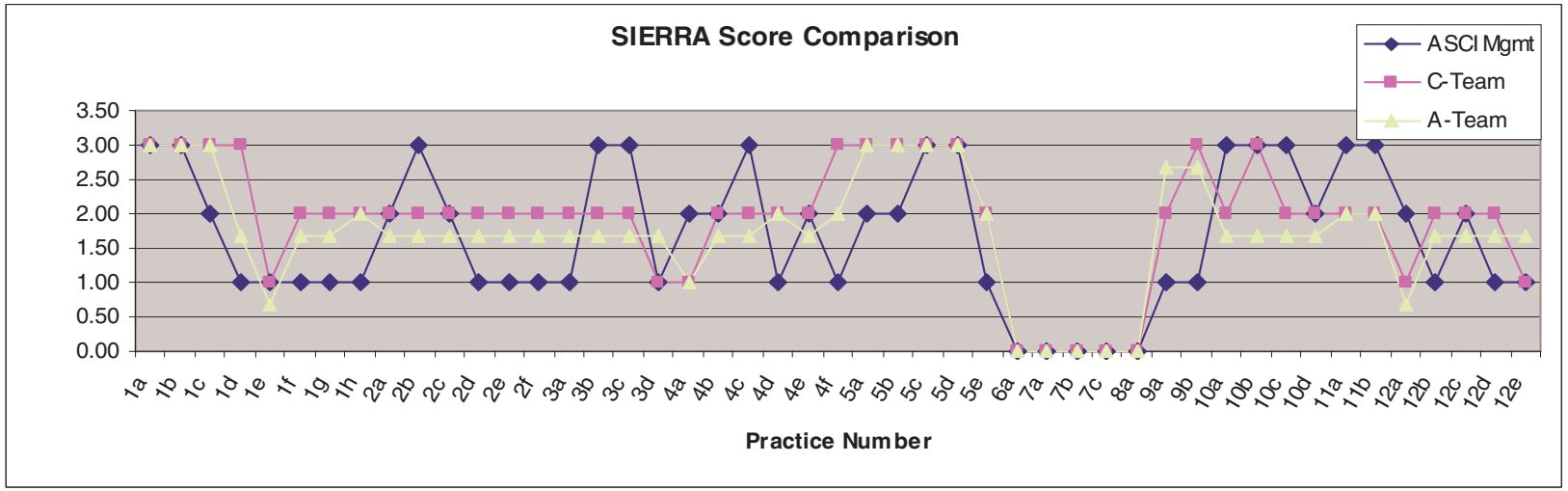

PRESTO Gap: A-team minus ASCI Required
2.50
2.00
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

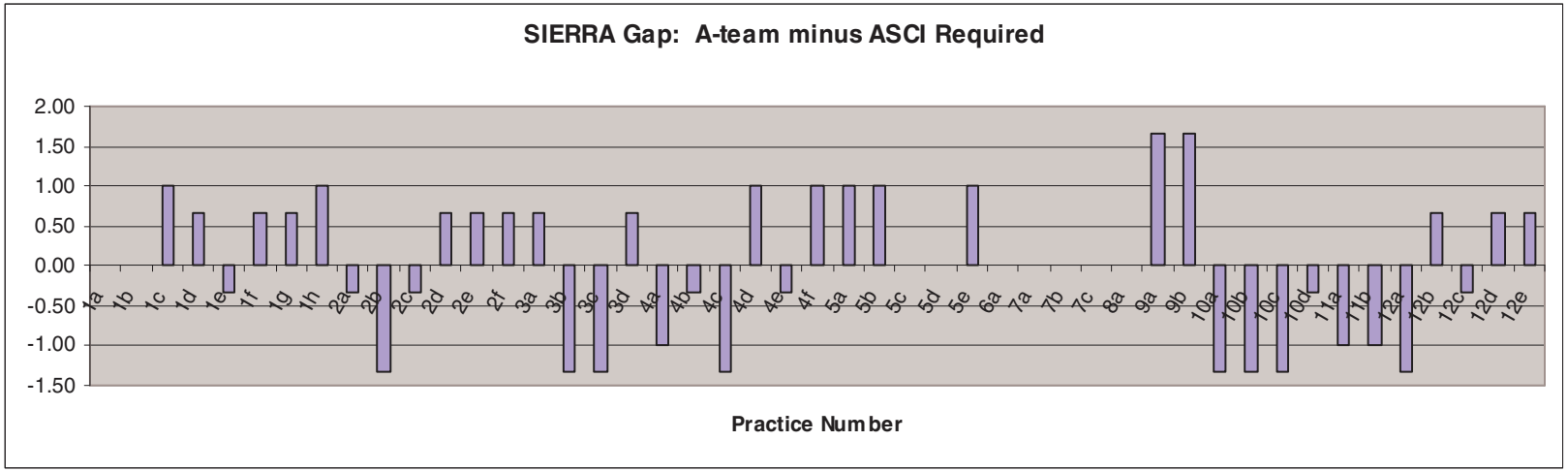
SALINAS


SALINAS Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

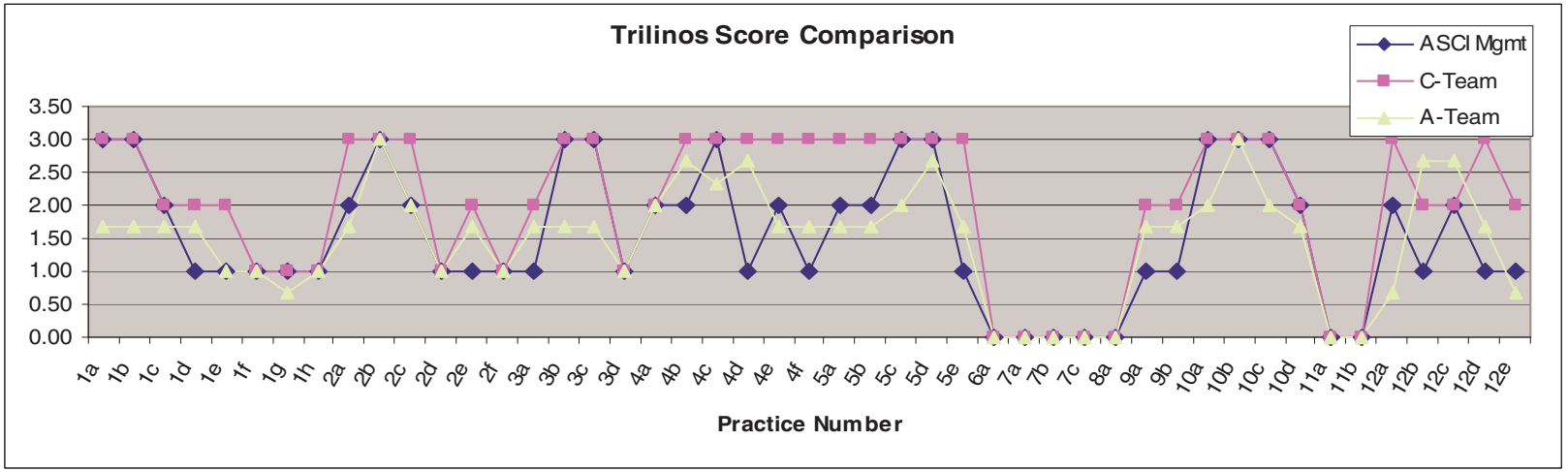

SALINAS Gap: A-team minus ASCI Required
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

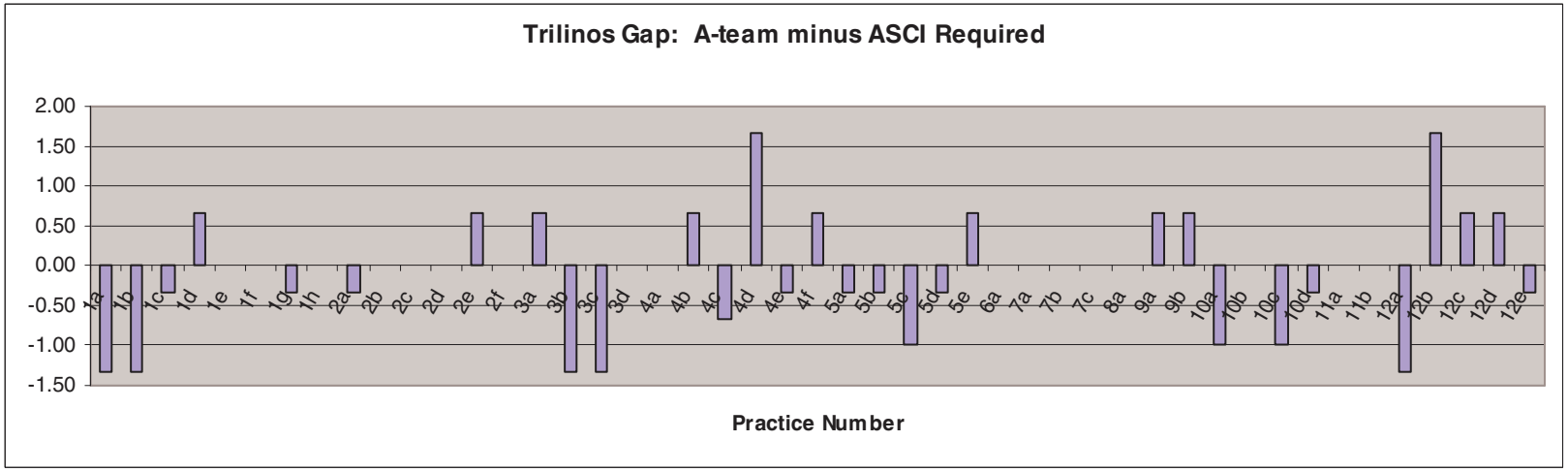
SIERRA Framework



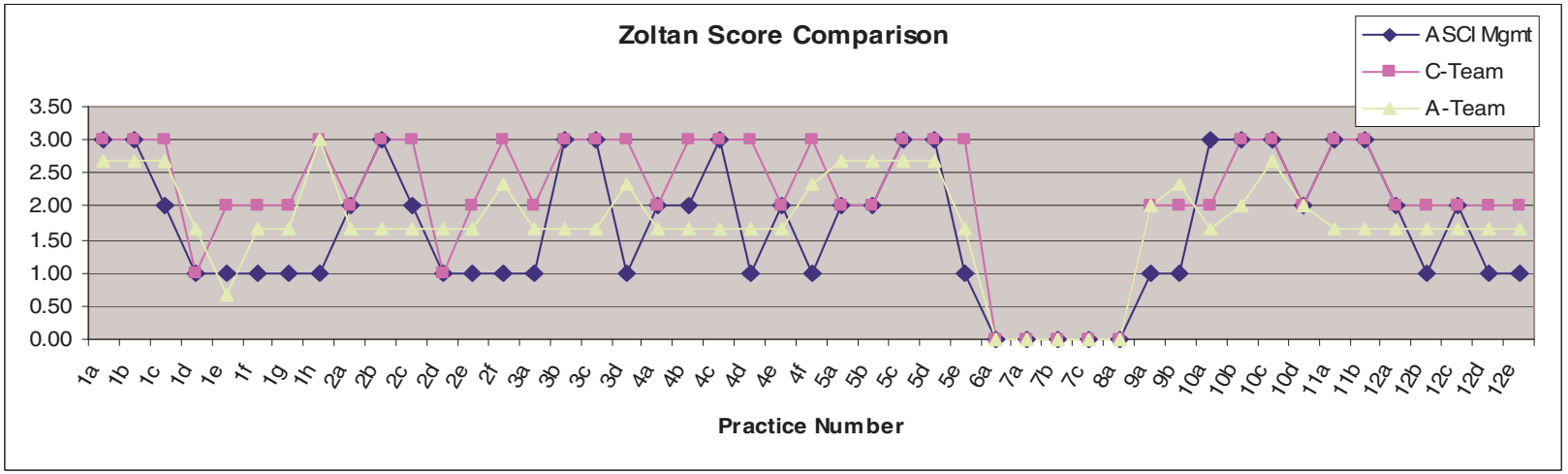


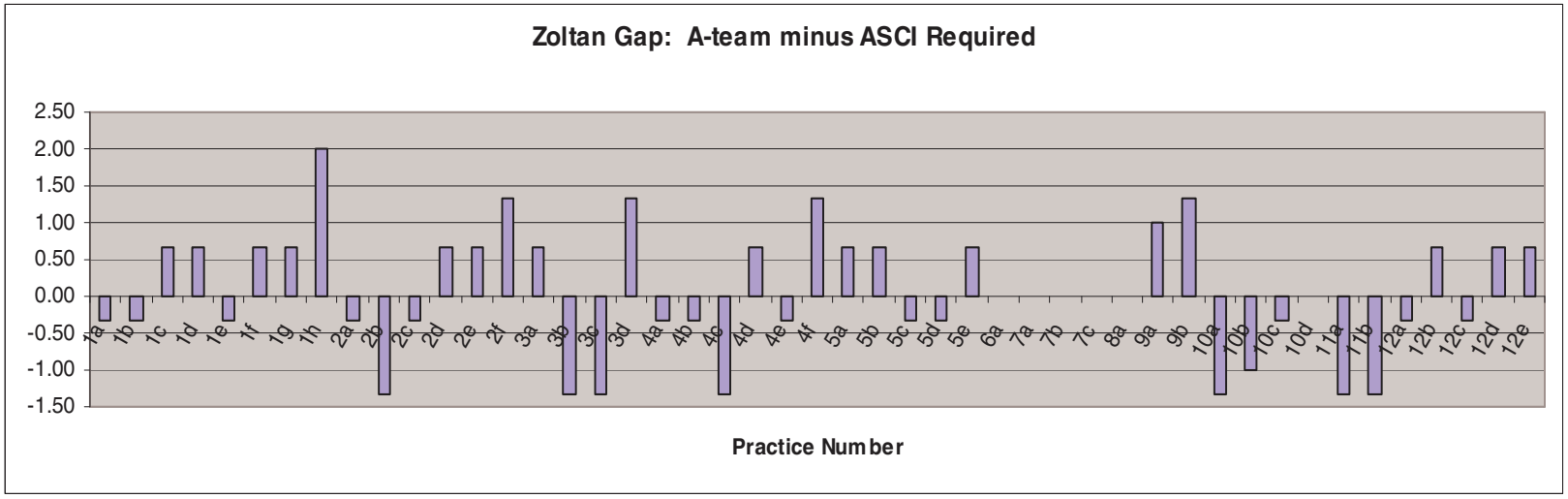
Trilinos


Trilinos Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number


Trilinos Gap: A-team minus ASCI Required
2.00
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

Zoltan


Zoltan Score Comparison
ASCI Mgmt
C-Team
A-Team
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number


Zoltan Gap: A-team minus ASCI Required
2.50
2.00
1.50
1.00
0.50
0.00
-0.50
-1.00
-1.50
1a 1b 1c 1d 1e 1f 1g 1h 2a 2b 2c 2d 2e 2f 3a 3b 3c 3d 4a 4b 4c 4d 4e 4f 5a 5b 5c 5d 5e 6a 7a 7b 7c 8a 9a 9b 10a 10b 10c 10d 11a 11b 12a 12b 12c 12d 12e
Practice Number

Appendix C Exit Questions Responses

Q1. What is working well in your organization?

(* -- indicates response applies to more than one category)

Category	Response
Team (15)	
*	Team organization and the way they communicate → high performance
	Team interaction
*	Small team with different focuses, all willing to learn
	Good team (small team), good vehicle for implementing one organization's math. models in another organizations applications
	Close proximity of the code teams offices
	Good team
	Team, motivated, can drive vision
	Mentoring relationships work well
*	Close interaction with customer and management; co-location with customers
	Team, overlapping responsibilities, staying ahead of customer curve
	Team collaboration
	Excellent small team dynamics
	Recruiting team members from university contacts
	Team is so capable; fun to watch when change occurs because they respond so well; close team
	General group dynamics
Communication (14)	
	More gets communicated informally than formally
	Interaction between teams focusing on critical issues (like getting parallel functionality from SIERRA framework)
*	Team organization and the way they communicate → high performance
*	Sharing of "better practices" across teams
	Good relationship with analysts (users)
*	Close interaction with customer and management; co-location with customers
*	Last assessment feedback for archival needs via website
	Team discussions
	Customer interaction
*	VIS people – interaction was wonderful. Very helpful
*	Requirements collection and review with customers
	User relationships and requirements
	Separate meetings focused on algorithms (provides valuable focus)
	Informal networks, customer relations via project leader
Tools (29)	
	Tools – seamless for team to use
*	Responsive to customers, folks are receptive to new tools
	Nightly regression testing
	Bugzilla/Bonzai/CVS tool synchronization
*	Testing
	Nightly regression tests
*	Find where the most value added is in the processes. Example – SourceForge as a tool provides records with a single tool. A big win in showing traceability
	SourceForge
	SIERRA Framework works well. The ability to develop complex message

Category	Response
	passing codes without developing message passing works well
	Automatic archiving of email and other material provides good reference ability
	Infrastructure in SIERRA is unprecedented – very broad and flexible. Allows focus on code teams' real issues
*	Testing and CVS usage
	Good third party tools
	Changes tracked in CVS
	SourceForge for archiving and info control
	Ability to track bugs and changes to code
	SNTools
*	Testing
	SourceForge for code repository, issue tracking, document archiving
	CVS, control and versioning
	CVS, SourceForge
	Can implement needed ideas, infrastructure is in place
	SourceForge for issue tracking and commit logs is a good process (it would be nice to have a connection between SourceForge and DOORS)
	Good Tools: SIERRA Framework, Development Tools Changeover
	CM, tools
*	Testing and its ensuring confidence; some nice tools
	Regression testing
	SourceForge as an archival and control tool
	Automated regression testing
Management (4)	
	Broader spectrum of algorithms and technologies (libraries and algorithms) are made available because of the ASCI Apps program. (Collective development of algorithms.)
	Workload
	Limited micro-management
	The way the project is run
SQE/SQA (31)	
	Most everything but especially the release process which is checklist based
*	Sharing of "better practices" across teams
	The team is doing a reasonable job of gathering needs and turning them into requirements
*	Testing
	Requirements tracing
*	Testing and CVS usage
*	Testing
	Structure for code development
	Fluid design cycle
	Test centric approach
*	Testing and its ensuring confidence; some nice tools
	Code development
	Testing
	Checkin, checkout, and test process and development cycle works well
*	User relationships and requirements
*	Find where the most value added is in the processes. Example – SourceForge as a tool provides records with a single tool. A big win in showing traceability
	Instituted a slightly more formal code review process (and requirement documentation in the code commit – name of reviewer – and emails to group)

Category	Response
	Policies on regression testing (protect code)
	Delivering (product) to customers with competing requirements; treating software engineering as important
	Some folks recognizing the link between SQE practices & project value without added overhead
*	Last assessment feedback for archival needs via website
	Upfront design
	Can do Extreme programming
	SourceForge processes
	Checking reviews
	Pre-checkin quality reviews
	Tight controls on TPL's
*	Requirements collection and review with customers
	Code is documented well
	Improved code development guide (encouraged by the SQE assessment process)
	Improvement that's value-added without adding overhead
SNL Organization (9)	
	SNL hires people with good background and the motivation to learn.
*	Responsive to customers, folks are receptive to new tools
*	Small team with different focuses, all willing to learn
	Support people (libraries, machines, day to day development issues) take their jobs very seriously
	World-class product
	Access to various other support personnel (parallel profiling, serial profiling, ASCI machine help…)
	Satisfying customer need, high-level advancement in technology, good people funded for this project
	As a user, I can state needs
*	VIS people – interaction was wonderful. Very helpful
Resources (2)	
	Access to "big" machines
	$'s allow some research that would not be allowed otherwise
Other (3)	
	Little training or push to use numerous tools across ASCI
	Code is level of maturity where it can be used for bigger problems (that have a real impact on providing answers to tests for weapons groups).
	Code meets needs; reduced time from for simulations from 2 months to 30 minutes

Q2. What is not working well in your organization?

(* -- indicates response applies to more than one category)

Category	Response
Team (3)	
	Contributions from transient team members
*	Team is a very small group: many things are implicit and formal processes can produce a screeching halt or at least a slowdown
	LTEs on team, other (regular) team member has too little development time
Communication (6)	
	Handoffs with other teams
	Need to get more acceptance test information and direction from the requirement owner(s)
*	Old distribution (SIERRA tools) process. Generated problem tickets that were distribution or execution bugs, not code bugs
*	Team is a very small group: many things are implicit and formal processes can bring work to a screeching halt or at least a slowdown
*	Communication between the rank and file and whoever is dictating what your final course of action is. (ex - worked with the group on what they could address - but they don't know what is due when - do not have access to that level of planning)
	Other teams change our source code once it's in production, and don't tell us
Tools (13)	
	New tools - loss of productivity and integration
*	Old distribution (SIERRA tools) process. Generated problem tickets that were distribution or execution bugs, not code bugs
	SNTools, aggressive project, took on too much
	Codes on different platforms, with different compilers and different options all contributing to non-productive time
	Frameworks provide an awkward fit for some codes
	TaskTracker needs work, some other tools, line (test) coverage
	Some tools not as user-friendly as they could be
*	Implementation / metrics for ASCI parallelization tools
*	ASCI red platform, swapping compilers that impact tool usage and code verification
	Framework structure - comes at too high a cost - development within this structure is impractically expensive. (Restated by another team member - Complexity of working within the SIERRA Framework is prohibitive for code development. Lack of adequate training and help from Framework group makes development take about 4X longer than it should.)
	Upheaval from unstable tools and changing platforms
*	Requirements tracking
	New tools & release, code distribution, both ad hoc
Management (20)	
	Management's view of supporting a release does not always include time for true user support. Users can not make best use of the application without this support
	Yearly offsite is now broke; fell apart this year
*	Software quality expectation / funding - DOE and Sandia interests: push towards short-term goals versus long term needs
	Software is changing so rapidly, we constantly re-invent ourselves. We should stay the course and not jump ship
*	Communication between the rank and file and whoever is dictating what your final course of action is. (ex - worked with the group on what they could address - but they don't know what is due when - do not have access to that level of planning)

Category	Response
*	Implementation / metrics for ASCI parallelization tools
	Milestone targets (and process) seems to change frequently (to satisfy external review panel) and that impacts teams' work flow
*	ASCI Apps has an incredibly focused scope on systems level and weapons related problems that does not work for all teams. Apps does not have much interest in the way of research
*	Constantly feel behind. Need to distribute the work to new people, not add to work of the same people
*	When you hire great people who are "pushing the envelope", can't expect them to stay the course, as well as perform QA
	Changing expectations
	Changing requirements, goals, and milestones makes life difficult for the application developers
	Developing and maintaining workable project plans. Teams are not doing as much (effective) project planning.
	More task-orientation to V & V
	Changing requirements and targets (program level)
	Expectation of straight-line spending
	Certain programs are over-managed (overhead is excessive for small projects)
	Changing milestones – clarity of new milestones
	Multiple funding sources with 6 different requirement sets to manage; some disconnect between the funding and using customers; communicating the importance of meshing
	Discontinuity between systems expectations and ASCI expectations
SQE/SQA (12)	
	Could do better with the way we trace requirements – resources are an issue
*	Team is a very small group: many things are implicit and formal processes can bring work to a screeching halt or at least a slowdown.
	Code documentation
	Generalized overarching C++ design
	SNL is good at building capability, but, support of release (customer support) is given less attention. Need an increasing emphasis on supporting users
*	Software quality expectation / funding - DOE and Sandia interests: push towards short-term goals versus long term needs
	Legacy basis of the code makes SQE more challenging
	Volunteered last year to help with SQE practices but never heard from anyone to follow-up
	Management practices
	Third party software definition
*	Requirements tracking
	Performance testing for codes
SNL Organization (4)	
	Continued concern that formality has a tendency to intrude and lower efficiency. Training is an example – informal is the way to go for this environment (fear of formalism)
*	ASCI Apps has an incredibly focused scope on systems level and weapons related problems does not work for all teams. Apps does not have much interest in the way of research

Category	Response
*	When you hire great people who are "pushing the envelope", can't expect them to stay the course, as well as perform QA
	No merit /reward for doing software development
Resource Issues (9)	
	The overloading of developers
*	ASCI red platform, swapping compilers that impact tool usage and code verification
	Shortage of technical writing expertise and records management folks
	Porting to new platforms (their state of readiness)
*	Constantly feel behind. Need to distribute the work to new people, not add to work of the same people
	Year-to-year funding cycles→ have to stop work, write funding proposals, go back to work
	Forces from outside the team → writing the annual IP takes away from code development
	Technology interest beginning to exceed research resources
	Too few people to do work; help desk too new to determine benefit
Other (6)	
	The complexity of the environment (a dynamic environment)
	Many pieces to learn; licensing issues; LAN support
	No general sense of a product
	Too many platforms to support
	Lack of formalism associated with training
	Learning curve and getting up to speed

Q3. If you had more $$$ where would you spend it?

(* -- indicates response applies to more than one category)

Category	Response
Team (0)	
Communication (3)	
	Documentation for building on variouis platforms
	Code documentation
	Document the framework - intent - and use of
Tools (8)	
*	One set of tools and experts for all of the tools (but not one-size-fits all approach)
	Regression testing across multiple platforms
	SNTools - document process description
	Tool automation, fewer simpler tools and training
*	SNTools - lack leadership
	Modularize our infrastructure; testing, user support
	Software tools, modernize hardware platforms
	SNTools
Management (2)	
	If more money this year, need to know the funding will be there later
*	SNTools - lack leadership
SQE/SQA (19)	
	Documentation
*	Requirements – faster
*	One set of tools and experts for all of the tools (but not one-size-fits all approach)
	Software Quality
*	Core S/W development skills to perform maintenance and update of old codes.
	Customer support, functionality, testing
*	More manpower to support development– they have lost people to attrition or other projects; need people to test and interface with customers
	SQE and evidence tracking
	Top down design for Rad transport (multiple groups doing Rad transport – LANL did a top down design)
*	Access (year round) to SQE resources for implementation and process review
*	Hire additional developers or support folks to provide better tool infrastructure support
*	Access to SQE (testing, requirements, …) knowledge and consulting
*	Someone to help others set-up their test environments
*	Testing person(s) to offload the project leader; employ “code breakers”
	Design to analysis issues that would make the code easier to use
*	Access to SQE resources
	Effort and focus on developing and carrying out V&V on a broader basis.
	More time on SQE
	More effort on standardization of low level SQE practices and anything that can be easily shared between projects
SNL Organization (2)	
	Research focus
	More research, publication of results

Resource Issues (37)	
	Hire more people to do some of the administrative work – the emphasis is on publishing but there is no time to do it
*	Hire additional developers or support folks to provide better tool infrastructure support
	Provide more manpower so they could maintain previous level of development
*	Requirements - faster
*	Access (year round) to SQE resources for implementation and process review
*	More manpower to support development– they have lost people to attrition and to other projects; need people to test and interface with customers
	Hire more qualified staff. Getting staff is not trivial
	More time on other platforms
	More coders for more features in software
	Coders
	More reliable machines than Janus
	Hire framework people to work directly with applications people (or to be a member of a framework/application dual team)
	Computer science folks to address performance and data structures
*	Access to SQE (testing, requirements, …) knowledge and consulting
	Testing person(s) to offload the project leader; employ "code breakers"
	Manpower--shorthanded—difficulties finding expertise
	If more money, would hire more people
	Support for milestones (people)
	Locate external resource with required skill sets
	Support for milestones (people)
	Locate external resource with required skill sets
	Core code development
	Core S/W development skills to perform maintenance and update of old codes
	More resources that would be available to all the teams
	More people
	Address platform development environment; more testing / testers
	Someone to help others set-up their test environments
	The right people (skills) are the problem, not $'s.
	Need more people to support the framework
*	Core s/w development skills to perform maintenance and update of old codes.
	More people to pursue new ideas
*	One set of tools and experts for all of the tools (but not one-size-fits all approach)
	SNTools - Framework qualified developers
	Get more people - resources to help with communications and coordination issues
*	Someone to help others set-up their test environments
*	Testing person(s) to offload the project leader; employ "code breakers"
*	Access to SQE resources
Other (8)	
	Electron trapping
	Code capability, low level LED (or LET?)
	High-altitude fire ball
	Have an internal advocate for the project
	User support
	Update and improve physics of the code
	Would ask customers, but probably new development
	Infrastructure could use help in gathering real needs from users and SNL community

Appendix D Personnel/Sandia's SQE Subject Matter Experts

(**NOTE: For the most current published list of these subject matter expert's please go to:**
https://wfsprod01.sandia.gov/groups/srn-uscitizens/documents/document/wfs079595.pdf)

Software Quality Area	Subject Matter Expert	Org/Phone
Capability Maturity Models (software and integrated)	Joe Schofield Laney Kidd Dwayne Knirk Patty Trellue	09510/844-7977 02662/844-1242 12316/844-7183 02900/845-9734
Configuration Management PVCS Users Group	Molly Ellis; Laura Lang Dave Peercy; Dwayne Knirk Mar McCornack Lora Bonano	09519/844-8258 12316/844-7965 06536/845-8719 09522/284-5057
Enterprise Architecture Data Modeling	Dave Cuyler; David Leong Molly Ellis	09519/844-6851 09519/844-8258
Model Based Product Acceptance	Perry Cowen	12326/845-7177
Programming languages; implementation	Gordon Dodrill	02662/844-7255
Project Management	Richard Sarfaty (PMIC)	09512/284-3487
Requirements Management DOORS Users Group	Kent de Jong	02993/844-1750
SILC - Software & Information Life Cycle process	Joe Schofield	09510/844-7977
Software Courses (sponsor; bring to SNL)	Linda Wilson	03021/844-8326
Software Engineering and Process Improvement	Laney Kidd Joe Schofield Karen Erickson (ICADS)	02662/844-1242 09510/844-7977 06521/844-9437
Software Metrics and Function Points	Joe Schofield [Certified Specialist] Dave Peercy	09510/844-7977 12316/844-7965
Software Product Acceptance	Dave Peercy	12316/844-7965
Software Quality Groups PEARLS Software Quality Engineering SEPG (Software Eng. Process Group)	Anne Hodges, Mar McCornack Mike Blackledge; Dave Peercy Joe Schofield, John Larson, Ray Trechter, Molly Ellis, Donna Eaton, Paul Merillat	06536/844-6284 12316/845-8307 09510/844-7977
Software Specifications	Dwayne Knirk	12316/844-7183
Software Tools Rational Users Group (RUG)	C. Mike Williamson John Ball, Jr.	06536/844-3792 06523/844-1356
Software Verification & Validation	Ann Hodges Gary Froehlich Mike Eckley	06536/844-6284 06536/284-3930 02661/844-4767
Testing; Test Cases; Test Suites	Dwayne Knirk Lorraine Baca Mike Eckley Jim Reitzel [Certified Test Engineer]	12316/844-7183 02661/845-9721 02661/844-4767 05853/284-4552
Software Quality Engineering Information Technology and Data Modeling High Integrity Software Systems Engineering Instrumentation & Systems Verification	Mike Blackledge John Larson Larry Dalton Lorraine Baca	12316/845-8307 09519/284-3311 02662/844-2520 02661/845-9721