IT-STAFF TRAINING: A PROGRAM FOR A RISING TIDE OF CHANGE

David McGuire
Associate Director, ACS
University of New Mexico
CIRT RM. 159
Albuquerque, NM 87131
dmcguire@unm.edu

Mary E. Hanson
AIS Training Coordinator
University of New Mexico
CIRT RM. 175
Albuquerque, NM 87131
mhanson@unm.edu

Computer Information and Resources Technology
University of New Mexico Albuquerque, New Mexico
CIRT Staff: 150 technicians
Administrative Information Support: 50 programmer analysts
3,000 Staff
1,507 Faculty
24,334 Student population

ABSTRACT

Information Technology (IT) professionals today must quickly learn new technologies, maintain skills in legacy technologies, and pursue professional growth.

The managers of an applications group are assisting staff confronting this challenge, by:
1. Dedicating staff to training coordination;
2. Developing cross-team standards;
3. Supporting a variety of training methods.

Goals are:
1. Improve staff ability to manage rapid technological change;
2. Bring new people quickly up to speed;
3. Reduce and manage training costs.

Training needs arise from: Tool and skill standards; project needs; and performance appraisals.

Training is delivered via: New Employee Orientation; “Just-In-Time” training; and cross-departmental training.
IT-STAFF TRAINING:
A PROGRAM
FOR A RISING TIDE OF CHANGE

What’s the challenge? Or, why do we feel like we’re drowning?

Information technology (IT) professionals have long prided themselves on their ability to learn new technology by simply reading the manual and staying up late. Training, if it happens at all, often comes after a system is delivered, or takes the form of “on-the-fly” tutoring given by whoever happens to know more than the “trainee”.

These learning strategies, while still useful and used, may not be the most productive use of a programmer’s time. In addition, they often add to an already-high stress level, due to the sometimes-overwhelming new environment in which we work. IT professionals today face:

- **Expanding tool-kit on all platforms.** There are almost too many choices for newer, better, cheaper technologies that become obsolete in shorter and shorter time frames. Gone are the days of proprietary solutions, one-vendor-fits-all. Every system component is market-driven to support multiple linkages, multiple platforms. As a result, some components that were once “black boxes” are now tools in their own right, with competing vendors and standards (consider network protocols!). Components that once fit neatly in single boxes (database, development language, spreadsheet, word processor) suddenly mutate into tightly integrated suites, or sprout new functionality to keep up with the competition.

- **Expanding number of supported platforms.** Not only are we developing systems on everything from the desktop machine to the departmental server to the mainframe, the network itself has emerged as a platform. The Internet and “intranets” have their own rules and rapidly-developing tools.

- **Increasing rate of technological change:** “Senior programmers” among us will remember how long you used to be able to get by just knowing, say, COBOL, JCL, and a reporting language. That is now about the number of tools introduced weekly. Vendors are in an ever-faster race to become and stay competitive: It’s not enough to have a great database, you also have to offer Web development tools, object-oriented programming, a Web browser, support for mobile computing, etc. The vendor “push” then shows up as product (or hype) which our users see on TV or at a trade show, and think that they can have today if only the Computing Center would “just hurry up a little.”

- **Increasing pressure on educational institutions** to compete for students, contain growth or downsize, be accountable to (legislators, taxpayers, donors, businesses, community groups,...). Many of these efforts require the Computing Center to start new projects and use new technology. For example, institutions are trying in various ways to:
  - Re-invent themselves (re-engineering processes),
  - Present themselves better (Web presence, voice response access),
  - Understand student trends more quickly (data warehouse, decision support systems),
  - Improve services (point-and-click access).

**Strategies for Solution: Where are the Life-Boats to Keep Staff Afloat?**

This complex set of problems requires more than one strategy for solution. Useful approaches include:

- **Improve workload management.** (Just say no!) Better time-tracking and reporting can be part of the solution, by helping us communicate better with clients, understand where our time is spent, and know how long projects actually take.

- **Hire and promote change-friendly people:** IT projects today need people who are flexible, eager to learn, constantly preparing for change, self-directed, team-builders. Every current technical skill will become obsolete. A learner will not.
• **Build customer partnerships:** Develop customer relationships into partnerships, to permit a common point of view toward priorities, staffing, and shared goals. A strong partnership can do more to eliminate a “backlog” than any priority list.

• **Improve project planning.** A consistent, repeatable project-management method can help reduce the level of chaos, and increase chances for project success.

• **Let go of “total knowledge, total control”**. People accustomed to understanding everything about a language or a platform may need to let go of that level of control. Decreases in development cycle times and the generally increasing rate of change mean that the product or system one is expert in today, will be quite different tomorrow.

• **Publish standards:** Standards may constrain creativity in some regards, but reduce considerably the time devoted to routine maintenance, freeing programmers for more productive and interesting work.

• **Develop staff through training:** Training alone cannot boost productivity and eliminate stress. It can however provide consistency of basic knowledge, reduce supervisor time spent “mentoring”, and increase people’s ability to successfully manage rapid technological change.

**Training as Part of AIS’ Solution**

AIS (Administrative Information Systems, the administrative application support unit within CIRT) has established a staff training program. Our efforts to date:

1. Dedicating .5 FTE and office space to training coordination and support;
2. Developing cross-team standards for skills, knowledge, and work habits;
3. Supporting a variety of training-delivery methods;
4. Provide both ongoing funding and training time within projects.

AIS supports 4000+ programs, 24,000 students, and 6000 faculty and staff, with 43 programmers and analysts. The scale of our program may need to be adjusted for smaller or larger budgets. However, we hope that the tools and methods outlined below will be useful to any information-technology group trying to “keep their heads above water.”

**How Training is Designed:  What Our “New Ship” Looks Like.**

**First, decide what training is needed.**

We decide training content based on three sources:

1. **Project needs.** New projects are often the channel through which new technology flows. A user need, coupled with available new tools, allows new applications and requires new infrastructure. Some recent UNM examples of project-driven change include: Electronic Admissions using new Web tools; Student Data Warehouse built in Oracle on AIX; On-line Cashiering built in Visual Basic and MS-Access.

2. **Departmental standards.** Once projects have established a new tool in production, it becomes something others may use, which may require training. We codify that via three documents:
   - Job Duties & Responsibilities chart (Appendix A): Covers all aspects of our work, and documents how that work changes from programmer to senior analyst.
   - Technical Skills chart (Appendix B): Shows what skills and tools are important, at what job level.
   - Probation Evaluation checklist for new employees (Appendix C): Guides supervisors and new employees through the six-month probation period.
   - Standards manual: Describes specific techniques, forms, conventions, etc. to shorten the learning curve for new employees, or for infrequently used tools.

3. **Individual growth and remedial needs.** Individuals may want to learn in specific areas, considering projects they want to work on, or based on professional growth goals. Supervisors may also want to guide individuals to develop certain skills for growth, or to come up to a certain performance level for their grade. These needs call for targeted training, within a certain time-frame.
Second, Commit Resources to Training.

AIS has committed Mary Hanson in the role of Training Coordinator, to lead New Employee Orientation, and to develop and implement needed training events. Mary organizes the training events and when needed she may provide training. Her time commitment to training varies, based primarily on the number of new employees. She also provides internal PC technical support, and takes on specific assignments to keep her skills current.

AIS has set aside a part of a “resource room” for training and orientation activities. New Employee Orientation takes place here, as well as one-on-one training and use of specialized single-copy tutorials. Training has a budget line, for equipment, software, trainers, books, etc. In addition, AIS management advocates for training in many quarters.

Third, Deliver the Training.

We deliver training in three modes, each with its own triggering event:

1. New Employee Orientation, triggered by a hiring process;
2. Cross-departmental training, triggered by new projects or technologies;
3. Just-In-Time training, to meet individual or small-group needs.

Delivery Mode 1: New Employee Orientation

New Employee Orientation: Goals
Orientation is small-group or one-on-one training, led by the Training Coordinator, following a formal syllabus of topics to be covered (Appendix D). The goal is to bring each new person up to a basic level of familiarity with departmental tools and skills, and to begin to integrate them into the “culture.” The program assists in identifying individual strengths and growth needs, and promotes early self-sufficiency. The program also begins the integration of new staff into our “culture.”

New Employee Orientation: Setup
Unusual to many workplaces, our new staff do not report directly to their new supervisors. Instead, we spend their first month introducing the new employees to the campus, the department and the workplace.

When new staff arrive, they report to the Training Coordinator. The new staff may be a number of one to three based on our experience. The participants may be full time staff or part time students -- the program is tailored appropriately. UNM provides a general half day orientation. Mary answers follow-up questions, coordinates obtaining computer accounts and security access, and guides the new team members on tours of the campus and CIRT, with appropriate introductions to their new colleagues.

New Employee Orientation: Process
Prior to new employee arrival, Mary moves the assigned PCs into a training room, and strips them down to the basic Windows environment. Training actually begins with the individual team members building their own PC hard-drive with everything that they will use in the department’s production work environment. These exercises also provide knowledge the new staff can rely on if they need to rebuild their PCs due to viruses, hard disk crashes, or other unforeseen problems.

During Orientation, we introduce concepts on a variety of topics and apply them to real work situations. Orientation emphasizes tools for problem solving to give the individual early self-sufficiency. The process helps create a sense of freedom for discussion, problem area identification, problem solving and skill building.
New Employee Orientation: Final Stage
After the first four weeks of working in the closely supervised orientation team, we move the PCs back to the employees’ regular desks, and the employees report to their new supervisor. The Training Coordinator documents her observations of the new employee’s skills and work habits for the supervisor, using the Training Evaluation Checklist (Appendix E). The new employee also evaluates the Orientation process (Appendix F). The new employee may at that time be “buddied” with a team peer or expert, may work on assignments alone or may start out with customer related contact with other team members. The supervisor then uses the Training Evaluation checklist, along with the Probation Evaluation Checklist (Appendix C), to create and manage a skill development and tracking plan for that employee.

Delivery mode 2: Cross-departmental training.

Cross-Departmental Training: Goals
This is typically single-topic training presented to a large cross-team group by internal or external experts. The goal is to assist in launching new technology, or to reinforce or advance current skills.

Cross-Departmental Training: Strategies
On-Site Classes: We either ask internal experts to develop and deliver training, or contract with local or vendor experts. We occasionally require attendance at on-site training when changing our work place configurations. This has included migrating to Windows, changing fileserver and networking environments, and changing database tools such as IDMS or Oracle.

Off-site classes, conferences: We often stretch scarce training dollars by sending a few to formal off-site training, and asking them to pass that training on to others. Often an informal technical forum is all programmers need to get started.

Vendor-scheduled electronic info-sharing. Occasionally, vendor-scheduled “phone broadcasts” (IBM, Gartner Group, etc.) provide useful information. We are beginning to use satellite downlinks (such as the Oracle channel) to deliver such training as well.

Cross-team technology groups: Representatives of production teams meet regularly in technology-specific groups, to identify and solve problems, exchange information, and pass on training to their teams on new releases and new procedures.

Delivery Mode 3: Just-In-Time training

Just-In-Time: Goals
“Just-in-time,” a concept borrowed from manufacturing, indicates delivery of products or supplies just before they are needed to do a job. Just-in-time training, then, delivers new skill information as close as possible to the project start-time. We deliver it to individuals or small groups, and it must be either easy to schedule, or available in interactive media when the learner has available time.

Just-In-Time: Strategies
We have approached this in several ways:

1. Scheduled courses offered by established providers: We often steer individuals with specific needs to UNM’s many related courses offered by Continuing Education, the Anderson School of Management, the Computer Science department, Human Resources, and others. This can be “just-in-time” due to the wide selection of courses and schedules. We encourage staff to take advantage of their tuition waiver benefits. In addition, TVI and other educational institutions often present courses related to our needs. (Note: We purchase “market-leader” software wherever possible, to increase the chances of obtaining local training and support.)

2.
Electronic training media: We have invested in computer-based training (CBTs) for regularly used software such as SAS and Oracle. We have purchased a number of training videos, and can scheduled viewing as needed. Videos are also available to take home. Most new software products contain tutorials and context-sensitive help.

3. User-scheduled conference calls: We occasionally call on the expertise of professional analysts (Gartner Group, Yourdon, Oracle) or other universities for answers on specific, narrowly focused topics.

4. Internal “training-as-needed”: When necessary the Training Coordinator or other members of the department develop specific training sessions for individual or small-group needs. We hold a monthly technical forum that is an informal “hot topics” discussion ranging from “how to’s” to “why not’s.”

How is the program working today? Are we sinking or swimming?

The whole department has recognized the positive effect of the Orientation program. Mary regularly receives “appreciations” from staff members and supervisors for training events and Orientation successes. For the first time in years, we are hearing “no more training right now, thank you!”, indicating we are reaching an appropriate level of information matched to the need.

Supervisors are able to free senior staff from the tasks of basic skills training. Team-specific training now focuses on advanced and specific tool use, and business function concepts. Supervisors have a better inventory of team skill levels. Clients now receive more consistent service because new employees are better informed. New employees are better prepared for on-call responsibilities, and are better received by teams because they represent more help, not more work.

It is difficult to measure the specific impact of the training programs, due to the number of changes we are making in our environment. However, there is a general recognition by staff that training efforts have helped them be more effective, and feel more a part of a team.

What Have We Done in Three Years?

Since the time we created the Training Coordinator position, we have provided new employee orientation for 18 full time staff and 4 students. We have scheduled or provided cross-team training for 50 in-house classes with attendance ranging from 10 to 20. We have set up three technical groups who meet regularly to focus on topics in the areas of IBM Production, Oracle and PC topics. For ongoing Just-In-Time training we have purchased computer-based training packages for Oracle, SAS and Microsoft products. We have scheduled over 12 “tech forums” for topics including SAS/IDMS, SAS products, e-mail, scheduling, Windows, printing, etc.

Could Your School Do It?

Our program started modestly. Our first New Employee Orientation barely touched on the topics that we cover today. It was two weeks long and we spent about a week on technical areas. Our initial goal was to give new staff a non-management point of contact who could assist in problem solving. Our training coordinator originally did more learning, PC trouble-shooting and less training than today. As the program proved successful it grew to the four week program we have today.

We have recognized several success factors for these programs:

- Get ongoing commitment of management to fund, ask for, insist on, talk up, and otherwise support training, especially in the face of application deadlines.
- Dedicate a focal point for training coordination. The staff assigned to this position may need to develop new skills and attitudes that are not common to a technical position. They will need to work with management to
become pro-active in supplying training programs. They will also need to maintain and build their own skills. They will need time to build the position to your departments needs.

- Use cross-team efforts to develop skill standards. This improves quality and increases buy-in.
- Ensure a commitment of project managers and staff to continuous learning, and follow-up work assignments to apply new skills. Skills learned are lost if not used.

The program works for us and can work for you.

**Surfing Into The Future!**

We continue to look for ways to improve the effectiveness of our internal training. Some options:

- **Cross-CIRT training?** It may be possible to improve the service and still reduce overall costs by extending training efforts beyond the applications area to include all of CIRT. Many skill needs are obviously very different, but we may find common ground.
- **Develop “mentor lists”?** With expanding tool-kits and distributed expertise following distributed systems, it has become less clear who is expert in what. We are considering a “mentor list” to structure ongoing knowledge-sharing and mutual support.
- **Curriculum on the Web?** The Web is an obvious location for re-usable training material, which would join our Standards manual, already in place.
- **Planning document?** We are developing a long-range (8-12 month) training plan to help integrate training with application needs and team “down times,” and to help reduce costs by lengthening the time for search and negotiation.

We have found that our training programs have indeed improved the consistency of basic knowledge, reduced supervisor time spent “mentoring,” and increased people’s ability to manage and grow with rapid technological change. We look forward to expanding our life-boat into a speed-boat!
Appendix A: Job Duties and Responsibilities Chart

<table>
<thead>
<tr>
<th>PROGRAMMER II</th>
<th>PROGRAMMER III</th>
<th>ANALYST / PROGRAMMER I</th>
<th>ANALYST / PROGRAMMER II</th>
<th>ANALYST / PROGRAMMER III</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TECHNICAL:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Under close supervision, receive clear, detailed and specific instructions for assignments. Work results and method are regularly checked.</td>
<td>Under close supervision, receive clear, detailed and specific instructions for assignments. Work results and method are regularly checked.</td>
<td>Under general supervision, receive general instructions for specific and ongoing assignments, and refers problems and unfamiliar situations to supervisor. Finished work and method are checked.</td>
<td>Under indirect supervision, receive objectives, priorities and deadlines for assignments and handles problems in accordance with instructions, policies, or accepted practices. Completed work is reviewed for appropriateness and conformity to policy.</td>
<td>Under general direction, receive overall objectives and resources for completion of projects. Completed work is reviewed only from an overall standpoint in terms of effectiveness in meeting requirements or expected results.</td>
</tr>
<tr>
<td>Read someone else's basic code in a language they are already familiar with and determine how and what the program is doing.</td>
<td>Read someone else's code in a language they are already familiar with and determine how and what the program is doing.</td>
<td>Read someone else's complex code in a language they are already familiar with and determine how and what the program is doing.</td>
<td>Read someone else's complex code and determine how and what the program is doing.</td>
<td>Read someone else's complex code and determine how and what the program is doing.</td>
</tr>
<tr>
<td><strong>WORK HABITS:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clearly expresses self in oral and written communication; has good listening skills. A believer in asking questions when in doubt.</td>
<td>Clearly expresses self in oral and written communication; has good listening skills. A believer in asking questions when in doubt.</td>
<td>Clearly expresses self in oral and written communication; has good listening skills. A believer in asking questions when in doubt.</td>
<td>Clearly expresses self in oral and written communication; has good listening skills. A believer in asking questions when in doubt.</td>
<td>Clearly expresses self in oral and written communication; has good listening skills. A believer in asking questions when in doubt.</td>
</tr>
<tr>
<td>Has the ability to work under pressure to meet target dates. Be flexible when there are schedule or priority changes and last minute requests.</td>
<td>Ability to work and communicate under pressure to meet target dates. Be flexible when there are schedule or priority changes and last minute requests.</td>
<td>Ability to work and communicate under pressure to meet target dates. Be flexible when there are schedule or priority changes and last minute requests.</td>
<td>Ability to work and communicate under pressure to meet target dates. Be flexible when there are schedule or priority changes and last minute requests.</td>
<td>Ability to work and communicate under pressure to meet target dates. Be flexible when there are schedule or priority changes and last minute requests.</td>
</tr>
</tbody>
</table>
## Appendix A: Job Duties and Responsibilities Chart Page 2

<table>
<thead>
<tr>
<th>PERSONAL TRAITS:</th>
<th>PROGRAMMER II</th>
<th>PROGRAMMER III</th>
<th>ANALYST / PROGRAMMER I</th>
<th>ANALYST / PROGRAMMER II</th>
<th>ANALYST / PROGRAMMER III</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is punctual and has regular attendance; has no interference preventing same.</td>
<td>Is punctual and has regular attendance; has no interference preventing same.</td>
<td>Is punctual and has regular attendance; has no interference preventing same.</td>
<td>Is punctual and has regular attendance; has no interference preventing same.</td>
<td>Is punctual and has regular attendance; has no interference preventing same.</td>
<td></td>
</tr>
<tr>
<td>Strive for high standards and improvement in all tasks.</td>
<td>Strive for high standards and improvements in all tasks.</td>
<td>Strive for high standards and improvements in all tasks.</td>
<td>Strive for high standards and improvements in all tasks.</td>
<td>Strive for high standards and improvements in all tasks.</td>
<td></td>
</tr>
<tr>
<td>Must show a desire for job stability.</td>
<td>Must show a desire for job stability.</td>
<td>Must show a desire for job stability.</td>
<td>Must show a desire for job stability.</td>
<td>Must show a desire for job stability.</td>
<td></td>
</tr>
<tr>
<td>Must exhibit a neat, clean appearance and appropriate dress.</td>
<td>Must exhibit a neat, clean appearance and appropriate dress.</td>
<td>Must exhibit a neat, clean appearance and appropriate dress.</td>
<td>Must exhibit a neat, clean appearance and appropriate dress.</td>
<td>Must exhibit a neat, clean appearance and appropriate dress.</td>
<td></td>
</tr>
<tr>
<td>Possess integrity, honesty and is able to maintain the confidentiality of information.</td>
<td>Possess integrity, honesty and is able to maintain the confidentiality of information.</td>
<td>Possess integrity, honesty and is able to maintain the confidentiality of information.</td>
<td>Possess integrity, honesty and is able to maintain the confidentiality of information.</td>
<td>Possess integrity, honesty and is able to maintain the confidentiality of information.</td>
<td></td>
</tr>
</tbody>
</table>
## APPENDIX B
### ADMINISTRATIVE INFORMATION SYSTEMS
#### TECHNICAL SKILLS CHART

<table>
<thead>
<tr>
<th>PRIMARY LANGUAGES</th>
<th>PII</th>
<th>PIII</th>
<th>API %</th>
<th>APII % *</th>
<th>APIII % #</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>COBOL</strong></td>
<td>Simple code</td>
<td>Struct/complex code</td>
<td>Struct/complex code</td>
<td>Struct/complex code</td>
<td>Struct/complex code</td>
</tr>
<tr>
<td><strong>SAS</strong></td>
<td>Simple code</td>
<td>Simple code</td>
<td>Complex code</td>
<td>Complex code</td>
<td>Complex code</td>
</tr>
<tr>
<td><strong>IDMS: DBMS</strong></td>
<td>N/A</td>
<td>Understand concepts</td>
<td>Master concepts</td>
<td>Master concepts</td>
<td>Master concepts</td>
</tr>
<tr>
<td><strong>IDMS: IDD</strong></td>
<td>Use simple functions</td>
<td>Use simple functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
</tr>
<tr>
<td><strong>IDMS: ADSC</strong></td>
<td>Use simple functions</td>
<td>Use simple functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
</tr>
<tr>
<td><strong>IDMS: MAPC</strong></td>
<td>Use simple functions</td>
<td>Use simple functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
</tr>
<tr>
<td><strong>IDMS: DBOL</strong></td>
<td>Use simple functions</td>
<td>Use simple functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
</tr>
<tr>
<td><strong>MS ACCESS</strong></td>
<td>Use simple functions</td>
<td>Use simple functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
</tr>
<tr>
<td><strong>SQL</strong></td>
<td>Use simple functions</td>
<td>Use simple functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
</tr>
<tr>
<td><strong>VISUAL BASIC</strong></td>
<td>Simple code</td>
<td>Simple code</td>
<td>Complex code</td>
<td>Complex code</td>
<td>Complex code</td>
</tr>
<tr>
<td><strong>CULPRIT</strong></td>
<td>Simple code</td>
<td>Simple code</td>
<td>Complex code</td>
<td>Complex code</td>
<td>Complex code</td>
</tr>
<tr>
<td><strong>JCL</strong></td>
<td>Modify existing JCL</td>
<td>Write new JCL: master syntax</td>
<td>Master concepts: design jobs</td>
<td>Master concepts: design jobs</td>
<td>Master concepts: design jobs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECONDARY LANGUAGES</th>
<th>PII</th>
<th>PIII</th>
<th>API %</th>
<th>APII % *</th>
<th>APIII % #</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MARKIV #</strong></td>
<td>Simple code</td>
<td>Simple code</td>
<td>Complex code</td>
<td>Complex code</td>
<td>Complex code</td>
</tr>
<tr>
<td><strong>SAS ASSIST</strong></td>
<td>Simple code</td>
<td>Simple code</td>
<td>Complex code</td>
<td>Complex code</td>
<td>Complex code</td>
</tr>
<tr>
<td><strong>ALPHA 4</strong></td>
<td>Use simple functions</td>
<td>Use simple functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
<td>Use complex functions</td>
</tr>
<tr>
<td><strong>SAS/IDMS</strong></td>
<td>Simple code</td>
<td>Simple code</td>
<td>Complex code</td>
<td>Complex code</td>
<td>Complex code</td>
</tr>
</tbody>
</table>
### APPENDIX B
**ADMINISTRATIVE INFORMATION SYSTEMS**
**TECHNICAL SKILLS CHART**

<table>
<thead>
<tr>
<th>PRIMARY TOOLS</th>
<th>PII</th>
<th>PIII</th>
<th>API %</th>
<th>APII% *</th>
<th>APIII % #</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSO</td>
<td>Simple functions</td>
<td>Complex functions</td>
<td>Complex functions</td>
<td>Complex functions</td>
<td>Complex functions</td>
</tr>
<tr>
<td>DISPATCH</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>LIBRARIAN</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>E-MAIL (PINE)</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>SYNCHRONIZE</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>JCLCHECK</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>ACF2</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>VERSION MERGER</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>ABEND AID</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>XPEDITER</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>WORD PERFECT</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>DOS / WINDOWS</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>ESP</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECONDARY TOOLS</th>
<th>PII</th>
<th>PIII</th>
<th>API %</th>
<th>APII% *</th>
<th>APIII % #</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFP</td>
<td>Simple overlay</td>
<td>Simple overlay</td>
<td>Complex overlay</td>
<td>Complex overlay</td>
<td>Complex overlay</td>
</tr>
<tr>
<td>FTP</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>LOTUS 123</td>
<td>N/A</td>
<td>N/A</td>
<td>Simple functions</td>
<td>Simple functions</td>
<td>Simple functions</td>
</tr>
<tr>
<td>FREELANCE</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
<tr>
<td>PROJECT WORKBENCH</td>
<td>N/A</td>
<td>N/A</td>
<td>Phase plan</td>
<td>Full plan &amp; dependencies</td>
<td>Full plan &amp; dependencies</td>
</tr>
<tr>
<td>SYSTEM ARCHITECT</td>
<td>N/A</td>
<td>N/A</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
<td>Working knowledge</td>
</tr>
</tbody>
</table>

%  A/P I, A/P II, and A/P III’s supervising others will have additional management skills detailed on D&R chart. (Appendix A)

*  A/P II, and A/P III’s will have greater depth and experience at each level.

#  Language is targeted for removal from AIS. Career path training not advised.

Not every position will require every skill. When a position requires a skill, it will usually be at the level specified. Any position may have some requirements above or below grade. Per Human Resources Department, at least 60% of duties and requirements should be at the level of the position.
Appendix C: Six Month Evaluation Checklist

New Employee Evaluation Checklist for 6 Month Evaluation Period Decision

Analyst/Programmer I

This evaluation checklist is for the AP/I position, as defined and outlined in the CIRT Job Duties and Responsibilities Chart. The intent is to use this list to verify that the employee has the skills and ability to perform at the specified grade of their position. Ongoing evaluation and reassessment will be important over the course of working the list and working with the employee. It may be necessary to have the new employee go through several iterations of a task assignment in order to reach the level of confidence required by the group leader. Document the results from this checklist in the employee’s six-month Performance Appraisal.

If the new employee can perform the following tasks at an 80 to 90% efficiency rate, then they should be able to become a productive member of our staff at the AP/I level.

Comment: Provide guidelines and definitions for each of the categories listed below.

1. Prepare and give a presentation to the project team

2. Create/Modify a production job stream

3. On-call skills

4. Conduct a user walkthrough

5. Programming Skills

6. Create/Modify an ADS/O dialog

7. Modify/Develop a COBOL program

8. Modify/Develop a SAS program

9. Modify/Develop a Visual Basic, Oracle, or Access Application
APPENDIX D: New Employee Orientation Syllabus

1. Campus Tour: Physical Location Of Buildings On Campus

2. CIRT -- Department Structure and tour

3. AIS WORK ENVIRONMENT

   HARDWARE:
   1. PC/mouse/printer: Do they all work?
   2. IBM 9121, RS/6000: our mainframes
   3. AIX: Our UNIX environment (e-mail)
   4. Need Logons & Passwords
   5. Printers
   6. Building issues: voice mail, safety, hours, locks, other security issues

   SOFTWARE: (See Technical Skills Chart)

   MY PC: How does it work and how can I rebuild the hard drive if necessary?
   1. autoexec/config.sys
   2. Windows 3.11/95 highlights
      - Building Program Groups & Icons
      - File Manager/Explorer
   3. load/run F-PROT
   4. Directory/Filename conventions
   5. Backup to floppy/file using MS Backups
   6. load network client: VLMs / Client 32
   7. Win 3.11 pre Windows logon for fileserver access -- what is a fileserver, its
      construction, drives, mapping, etc.

   COMMUNICATION TOOLS
   1. Pine e-mail, FTP
   2. QVT -- home dial-up access (non-GUI)
   3. TN3270
   4. WinZip, Viewer
   5. LPR
   6. Synchronize scheduler
   7. Netscape -- intro to UNM home pages, on-line docs, UNM courses
   8. Fileserver Software

   IBM TSO/BATCH SOFTWARE: Getting our main job done.
   1. TPX -- multi-tasking in MVS environment: MVS windows!
   2. ACF/2 -- security concepts
   3. ISPF introduction or review:
   4. Production/ Test -- rules & standards of Job Cards and basics of JCL & dataset
      naming & migration procedures (more Netscape)
   5. Procs & Decks -- concepts & definitions & standards (more Netscape)
   6. TSO Add-on Tools: Overview, coding and use

   IBM IDMS Environment
   1. Introduction to IDMS
   2. Batch IDMS
   3. On-line IDMS
   4. Migrations from test to production

   ON-CALL PROCEDURES AND RESPONSIBILITIES

   DOCUMENTATION: What’s available, where and how to use it
Appendix E: New Employee Evaluation

New Employee Orientation: Evaluation Checklist & Trainer’s Turnover Comments

Date:_______
Trainer’s Name:_____________________________________________________
Employee’s Name/Position:____________________________________________
Group Leader’s Name/Group:___________________________________________

To Trainer: Considering your experience working with the employee during the orientation period, please answer the following questions and provide evaluation comments as necessary.

Orientation Worksheet Sections:
1) During the “PC” Section of the orientation, how well did the employee grasp the concepts covered in each of the sections (scale of 1 - 10, N/C= Not Covered)?:
   • _____ My PC:
   • _____Windows:
   • _____Unix Server/Windows Client/Desktop Suite -- Mirada:
   • _____Fileserver:
   • _____Analysis/Design Tools LAN versions:
   • _____Application Development Tools LAN versions:
   Did you have to unduly repeat any sections or material due to a lack of understanding?_________
   Additional comments pertaining to this section of training:_______________________________

2) During the “IBM” Section of the orientation, how well did the employee grasp the concepts covered in each of the sections (scale of 1 - 10, N/C= Not Covered)?:
   • TSO: ______________
   • IDMS: ______________
   • UNIX: ______________
   Comment: Each orientation section is reviewed.

Work Habits:
Communication skills: Oral_________Written__________Listening__________
Ability to work under pressure: ______________Ability to be a team player:_____________________
Self Motivation: ______________Ability to organize time:_____________________

Personal Traits:
Punctual w/Regular Attendance (employee on time for training?) ______________
Strive for high standards: ______________Desire to keep job:_____________________
Integrity/Honesty: ______________Neat and Clean Appearance:_____________________

Learning and Thinking Skills:
Comprehension ability: ______________Focus:__________________________
Focus: ______________Problem Solving:__________________________
Logic: _______________________________________________________________________

General:
How well do you think the employee will perform in their position. Focus on the position in general not just the specific team currently targeted for: __________________________________
What do you see are their strengths: _____________________________________________
What do you see are their weaknesses: ___________________________________________
How well do you see the employee fitting in the department? _______________________
In order to focus on the employee’s weaker areas, and bring them up to speed, where do you think the employee’s first assignments should be: ________________________________________________
Did you get a sense of the employee’s overall technical skills (e.g., COBOL, SAS, Visual Basic, Graphics, Word Processing, Spreadsheets, etc.)? Please speak to each language or skill you have information on: ______________

Additional Free Form Comment Area:
____________________________________________________________
Appendix F

**New Employee Orientation: Content Evaluation Checklist**

**Date:**

**Trainer’s Name:**

**Employee’s Name/Position:**

**Group Leader’s Name/Group:**

To Employee:

Your evaluation will help us improve the content, delivery and topics covered during the next Orientation. Please turn in each area as you complete it. Remember that this training is for you to gain as much information as possible and your feedback can only improve the process for yourself and others to follow. Be sure to review your handouts and notes before answering the questions.

Considering your experience working with the trainer during the orientation period, please answer the following questions and provide evaluation comments as necessary.

**How did you find the campus tour:**

1. Extremely useful   2. Interesting   3. Okay   4. Not Useful for me specifically   5. Don’t do this again

**After we completed the information about CIRT, are you able to:**

find your way to (name the area): _____________________________

(May request an evaluation of each topic of the orientation.)

**General:**

Do you think that this orientation may assist you in performing your job? ____________________

What do you see as the most valuable part of this training: _____________________________

What is the least likely part of this training to be used: _____________________________

Considering this orientation, what do you think your first assignment should be doing: (JCL, batch process work, IDMS, PC applications, project planning, etc.)__________________________

**Additional Free Form Comment Area:** _____________________________

_______________________________________________________________________________