Taking it to the next level:

Implementing Your Roadmap

You have assessed your Business Requirements…

You have evaluated your operational capacity…

You have a plan for systems technology…

You have chosen a provider…

… Now the real work STARTS!

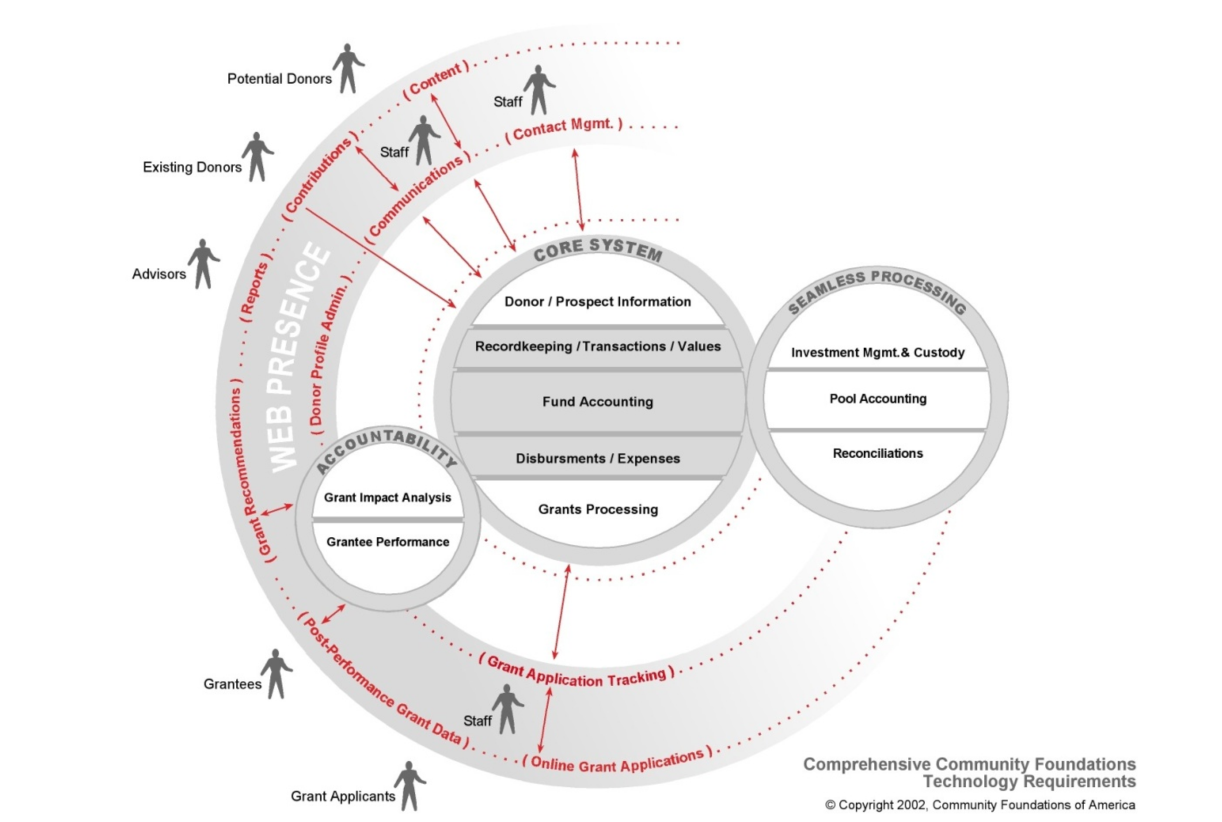
TABLE OF CONTENTS

This document was prepared by Douglass Yeager of the Community Foundations of America (dyeager@cfamerica.org) as one component of his project support to the Technology Steering Committee. (10/10/2006).

1. Introduction:

What does “application landscape” mean? What does it look like? Four years ago the TSC proposed a bold new vision for foundations named the “Technology Roadmap” (see below). This roadmap began with the core systems associated with foundation back office processing, but then asked the simple question: “How will our clients relate to this system?” How will this system seamlessly operate with our investment custodians? How will we extend the functionality of this system?

Figure 1: The Technology Roadmap



A. *The Current Landscape*

In research performed within the TSC about the applications landscape, several salient points emerged:

- ✓ Specific solutions are now available, but some solutions are more “available” than others.
- ✓ There is no “one-size-fits-all” solution. Any one foundation will have a unique product mix. Community foundations are a diverse group of organizations of differing size, product mix and perceived community role.
- ✓ Each applications area has several flavors of solutions available.
    - o Within one line of business, (e.g. competitive grants) some community foundations will do it when asked, others will consider it a part of their stock and trade, and still others will consider it a strategic emphasis. Differences in scale matter.

- A foundation's needs change over time: a particular solution that made sense one year may no longer be a good fit the very next year if it is not aligned with changing foundation priorities.

Three tools are included in this guide to implementation to help manage this milieu:

- Appendix B: Ongoing Feasibility Assessment – The high level feasibility assessment in the appendix will clarify ongoing issues and identify areas that will need focus in order to move towards implementation.
- Appendix C: Getting Ahead of Change – This appendix will give insight into the process of Evolving the Roadmap, projects that will need to be done in 3-4 year horizon to further scale a foundation's processing.
- Appendix D: Outcome/ Solution Matrix –This matrix is based on the reality that there really are a finite number of tools (listed across the top) available to meet a diversity of foundation needs (listed down the side). This graphic provides a bird's eye view to the state of the landscape.

2. Implementation Approach

A. *Definitions*

Implementation is getting a project up and running. Transition is turning the product over to the people responsible for using it. This Implementation and Transition Plan document describes the steps necessary to turn on a system and turn it over to its many ultimate users. The plan assures that all of the necessary steps are identified and that each of these steps has resources assigned to them.

Implementation and Transition planning is important because it pulls together the tasks necessary for placing a system in the hand of its users and customers. It includes elements that will facilitate communication and coordination across the team for a successful rollout. Managing these challenges well in advance of the implementation/transition date help make the transition smoother.

This document will focus on three aspects of implementation planning:

- Resource Requirements – Making sure you have all of the resources (financial, staff and management) to have a successful rollout. This section includes detailed templates for task assignments.
- Best Practices in Implementation and Transition – What are the things you need to watch carefully?
- Training - The proof is in the pudding: if users don't use the system, it is a failure. But training is expensive. This section includes strategies to formulate a cost effective just-in-time training plan

These templates and best practices will smooth the process of implementation and transition. Planning and communication are key success factors.

It is critical to remember while planning that most staff have day jobs. Just because a task is well defined and in and of itself achievable, when they add up one on top of another and on top of "regular" work they become impenetrable. Change is hard, and people need a special task inserted into their schedule: rest. This is an antidote to burn-out and errors, and since relationships suffer under stress, it is also healthy for the group process.

3. Resource Requirements

It is essential to not only understand the nature of what a project looks like, but also to know the resources to be used by an organization in order to implement it.

A. *Budget*

Having a financial plan for a project makes a simple statement: "this project is worth real money". This engenders an instinctive insight into ROI, and provides some measure should things get out of control.

In smaller organizations this process is particularly difficult. One approach is to draft a project budget that incorporates existing resources. For instance, a department can assign one fulltime equivalent to the project for its duration.

The TSC white paper "Five Steps to Budget Planning for Your Custom Roadmap" sets forth major budgeting principles. The spreadsheet model that comes with it actually will assist in providing estimates for line items.

B. *Schedule*

Three elements make up the implementation schedule:

- Outside constraints: factors such as grant cycles and year-end processing.
  - A separate document should list these outside constraints. It is not a long list, but it will be part of a critical risk management early warning system.
  - On a monthly basis, during project status meetings, this list should be reviewed to make sure something in the environment hasn't changed that will affect the project.
- Delivery Milestones: things users will notice. This starts with a Statement of Work (SOW) document establishing project expectations that is shared with all stakeholders. An SOW should be part of any project, whether or not there is an external vendor/ consultant.
  - The milestones should be few in number, but enough to provide to the organization a sense of progress through the project.
- Project Management (PM): tasks related to implementation itself.
  - Task tracking is central to PM: it can be done using simple document templates or sophisticated enterprise-wide databases, but the core function remains the same: keeping track of detailed tasks, who they are assigned to, and when they are due.
  - A "best practice" is to keep at least two task lists going: one at an executive level dealing with higher level issues, and the other with more pedestrian tasks. Executives just get confused by the detail, and high level issues make people in the trenches anxious.

All schedule related documents are living documents (i.e. need to be updated).

C. *Roles & Responsibilities*

The final key resource question relates to the need to identify the roles and responsibilities associated with implementation, including the skill set needed to perform those functions. Key roles to identify include the primary project manager, implementation team members, key technical staff (including the vendor), help desk support, and documentation.

A detailed tracking list for project implementation is provided in Appendix A: Implementation Tasks Control Document. Note that this list does not have any new feature or function development. If the product is not ready to be used, it is not ready for implementation. Scope control is paramount.

This detailed list is reduced below to a checklist overview (an "executive" version) to control to the longer task list shown in the appendix. Note that defining key interactions is central to this checklist: a project will be successful when stakeholders are involved.

Figure 3: Implementation/Transition Checklist	
Have dates been applied to all tasks?	☐
Is there an assigned person who is responsible for completing each task?	☐
Have dependencies between tasks been identified and communicated?	☐
Has the plan been reviewed with all impacted stakeholders and resources assigned?	☐
Has the project schedule been reviewed and updated?	☐
Is a dry run or tabletop exercise planned or required?	☐
Is a transition plan in place to get user signoff before turnover?	☐

4. Best Practices in Implementation and Transition

Transition planning would be incomplete without a discussion of how to turn the system over to the people who will be using it. This step is frequently overlooked in smaller organizations, but it is a critical success factor. While it is too early to map out precise transition milestones on a project plan, these need to be done!

Why do this?

- Change is scary.
  - People who are scared and have no power are called victims.
  - Victims will act out in any way they can – but negatively.
    - This system is about doing things better.
- Performing a formal turnover tells your users that you respect and value them.
  - It is the customer (the user) that signs off on that a Test Plan has been successfully completed according to pre-determined test criteria.
  - It is the user that judges documentation & training.
  - Notifying customers of the implementation date well in advance allows them to plan accordingly.

When the user feels in control not only will it eliminate negative behaviors but it will promote the kind of engagement and problem solving that builds success.

Once the system is ready to be used, a formal turn over process to the user community is an essential ingredient to user acceptance. Once the functionality and reliability of the product is shown to meet the acceptance criteria and the users buy off on it, the ownership of the system is transferred to them from the project team.

The following best practices are adapted from a leading public organization:[1]

- Perform a formal turnover whether the project uses a phased approach or not.
  - Conduct a walk-through of the Implementation and Transition Plan with stakeholders right before execution of the plan.
  - Obtain formal buyoff from the customer tha t a Test Plan has been successfully completed according to pre-determined test criteria.
  - Complete product documentation in advance of transition.
  - Establish a baseline of the product or service at the time of turnover.
  - Notify customers of the implementation date well in advance.

- Plan for a bumpy ride:
  - Implement the system or product on a smaller "pilot" scale to lessen impact to the business area in the event problems are experienced.
  - Establish contingency plans to recover if the system/product fails upon cutover.
  - Provide additional support staff when implementation is at full cutover.

- Keep the customer extensively involved in the deployment so they are aware of any potential disruptions to the production environment.
  - Provide a convenient and publicized means (e.g., centralized Help Desk) for customers to report problems to the appropriate party (e.g., vendor, project manager, system administrator, etc.).
  - Communicate status to customers, project team members, and stakeholders on a frequent basis so changes are expected and prepared for.

- Ensure business and technical support staff receive training as close to the implementation of the product as possible (just in time training).
  - Include specifics about transition to in-house staff if a vendor is involved in the initial system implementation.
  - Identify and train in-house staff so knowledge transfer is effective.
  - Plan for staffing levels and other facility or resource issues.
  - Document all commitments to stakeholders that maintenance staff will be expected to honor.

- Conduct project closeout activities prior to transition.
  - Utilize lessons learned to help in planning for next phase.

Document outstanding issues, problems, and change requests so that project staff will have a clear understanding of the product at the time of turnover. Having a team member assigned to the role of “pack rat” can really help.

5. Training

A. *Project Management*

The central theme the above *Best Practices in Implementation and Transition* is that an organization cannot spend too much time communicating with its stake holders about an important project. Well executed training is foremost on the list.

- The critical objective in training to deliver the training to users soon enough to prepare them for their new duties (and overcome change related anxiety), and not

[1] Washington State Department of Information Services, ©2004, http://dis.wa.gov/index.htm

so soon that by the time the system is in production they have forgotten what they learned.

- Another critical objective is to keep training costs under control.

Several strategies are proposed in the following pages to deal with the substance of these objectives and to develop a training plan.

- An outline of what is being installed offers (key skills areas)
- An outline of the foundation business requirements being met.
- A departmental level perspective on implementation priorities (as in, who should go first)
- An inventory of actual training resources.

It is important to view training as its own sub-project to insure an effective program. The checklist below provides a control to make sure that this sub-project is completed:

Figure 4: Training Plan Checklist	
Has the training audience been identified?	☐
Have training objectives been identified?	☐
Has funding for training been included in the project budget?	☐
Have the tasks in the training plan been integrated in to the project schedule?	☐
Have the missing skills been identified and will they be addressed in the training?	☐
Have the training needs been distinguished from the training wants?	☐
Has the training been scheduled?	☐
Have the training materials been prepared?	☐

B. *Platform Functional Overview*

The first step in understanding training requirements is to understand the system being adopted. These platforms have evolved to meet the diverse needs of its nonprofit clients: after listing them out you may realize that entire modules are irrelevant. Few vendors are able to provide a simple list to you.

The simplest way of developing a list of functions is to assemble all of the training manuals available from the vendor and cut & paste their tables of contents into a separate document. Chances are that the highest level categories will give you words to associate to a function (e.g. Event Management). You are not interested in the detailed “how to’s” (such as “Create new event”), so delete them.

C. *Business Needs & Platform Functions*

The critical question for training is: what skills do staff need in order to use the system? How can we avoid unnecessary training? How can we make it as relevant as possible?

The answer lies in understanding the foundation’s business needs and overlaying these with the system. To illustrate, envision a core CRM function (Contact Management) through a series of specific foundation business needs. This matrix will help determine which staff (related to the business need) will require training in this element. Each role within the system will touch different tools to get its job done.

Figure 5 Vendor Functions/ Foundation Requirements

Foundation Requirements ⇨ / Vendor Function ⇩	Data Integrity	Centralized View	Build Expanded Donor Profiles	Online Giving/Pledges	Document Management	Task Tracking	Special Event Tracking	Scholarships	Tracking grantees:	Marketing Communications
Contact Mgmt										
1. Individual Contacts										
2. Organization Records										
3. Managing a Contact										
4. Unit Contact Record.										
5. Relationships										

D. *Prioritizing Training*

The next step will be to understand the roles envisioned within the foundation for the platform and so reduce the training burden on individual staff members. It is fine to know what skills go where, but which roles are more important? Which need to take priority in training?

The prioritization of training requires a **departmental** perspective and participation in the priority setting. Hard dates, such as year-end giving, will drive this schedule. Any opportunities for pilot deployment of a subset of functions to a narrow audience should be explored for early adoption.

Planning for a flexible training program (as opposed to a brute force train-everyone-at-the-same-time), will save resources and provide more relevant skills at the right time for users.

E. *Training Summary*

This section has provided an analysis of the training needs from the perspective of the nature of the vendor platform, the business needs of the organization, and the sense of priorities within the organization.

The final step will merge this plan with actual training activities. These will include traditional in person onsite and offsite instruction. However, outside of general purpose introductory sessions, these methods are expensive and difficult to time effectively.

New training options should be explored with the vendor. These include:

- Distance Learning Technologies: use of online discussion technologies allows for both group and individual instruction. The great advantage of this technology is the flexibility they offer: classes can be as short as an hour and therefore address only topics of current interest. This arrangement obviously also reduces travel time for the instructor.
- Adult Education Models: Self Paced Tutorials: experience in providing higher education opportunities for working adults has demonstrated the effectiveness of self paced instruction, especially when used in conjunction with distance learning techniques. The knowledge that at the end of the week there will be a meeting with the instructor is an incentive to get it done.

The training plan will include a detailed program for each employee.

II. Appendix

1. Appendix A: Implementation Tasks Control Document

Task Description	Resource Assigned	Date Due	Comment	Status
Obtain Approval to Implement				
Finalize Implementation Plan				
Review implementation plan				
Approve implementation plan				
Conduct Dry Run Test of Implementation Plan				
Prepare test environment				
Conduct dry run test				
Notify Users of Implementation				
Prepare notification document				
Notify customers				
Prepare Production Environment				
Determine Configuration needs				
Establish network connectivity				
Prepare Training Environment				
Identify needed training facilities, tools and equipment				
Schedule sessions				
Invite trainees				
Refine training materials				
Make sure there are enough logon ids for training				
Test accessibility to system from training site				
Conduct training				
Install Production System				
Define and establish needed security at all levels				
Convert data				
Release to production and staff				
Update Project Control Documents				
Lessons learned session and documentation				
Additional Project Closeout Tasks				
Complete inventory of documentation				
Facilitate and transfer of knowledge that needs to occur				

2. Appendix B: Ongoing Feasibility Assessment

Feasibility Assessment		
Summary	Foundation:	
	Provider:	
	Solution	
Time	Proposed	2006 Q4 __ 2007 Q1__ Q2 __ Q3 __ Q4 __
	Other priorities During period	a. ___ b.
Cost	First Year License	$
	Configuration & Training	$
	Customization	$
	Ongoing License/Maint	$
Benefits	Problems to be solved	a. ___ b.
	Opportunities to be gained	a. ___ b.
Effects	Impact on work processes	a. ___ b.
	Training needs	a. ___ b.
Project Mgmt	Who and when?	a. ___ b.
	External resources?	a. ___ b.
Risk	Organizational capability	a. ___ b.
	Technology & Interface risk	a. ___ b.
	Risk in NOT doing solution	a. ___ b.

This final page may be copied for multiple providers.

3. Appendix C: Getting Ahead of Change

Wouldn't it be nice if there was a framework that could help anticipate change? That could help understand the issues involved with scaling philanthropy? That would lead directly to a process stepping through practical answers?

It so happens that such a process does exist. The approach builds on work done at the Software Engineering Institute, Carnegie Mellon University and at IBM around 1991, to "describe the software engineering and management practices that characterize organizations as they mature their processes for developing and maintaining software."[2] The result was known as the Capability Maturity Model (CMM).

What does software development have to do with foundation operations? The genius behind CMM is that it was really interested in how to achieve quality process – and the most thorough data available anywhere on "knowledge worker" activity happened to be in the development of software. Back then it was not unusual to for a company to have a thousand programmers.

The basic concepts of CMM have been applied widely in the last 15 years as organizations large and small have struggled with many of the same issues facing community foundations. CMM concepts can be found underlying such programs as ISO/9000 and Six Sigma. One of the latest such efforts is known as COBIT.[3]

From its executive summary:

> *"The key question is whether an enterprise's investment in IT is in harmony with its strategic objectives (intent, current strategy and enterprise goals) and thus building the capabilities necessary to deliver business value. This state of harmony is referred to as "alignment." It is complex, multifaceted and never completely achieved. It is about continuing to move in the right direction and being better aligned than competitors."*

Does this sound familiar? They study best practices, establish standards, use plain language, but best of all do so in a way that understands that one size does not fit all. For instance, if I am cooking for my family, or for a Church supper, or even for a large restaurant, it does not make sense to make soup the same way as a state of the art *Campbell's* soup factory.

The following page contains a chart that describes some attributes of processes at the 5 different levels of CMM, (Initial, Repeatable, Defined, Managed, and Optimized).

- Think of a process you currently do.
- Look the chart over and find where you fall in a given category.
- Now move down one cell – this describes what your next step needs to be.

The **Evolving Roadmap** adapts this chart to specific community foundation issues. It provides insight into not just what a foundation needs to do next year, but what it will be doing 3-4 years down the road, i.e. re-doing a current project to take it to once again the next level..

[2] Capability Maturity Model for Software, Version 1.1, Paulk, Mark C.; Curtis, Bill; Chrissis, Mary Beth Chrissis, and Weber, Charles, Software Engineering Institute, CMU/SEI-93-TR-24, DTIC Number ADA263403, February 1993.

[3] COBIT: Control Objectives for Information and related Technology, see Information Systems Audit and Control Association (ISACA®) http://www.isaca.org

Table: Maturity Model Attributes						
0 Non-Existent	**Understanding & Awareness**	**Training and Communication**	**Process and Practices**	**Techniques and Automation**	**Compliance**	**Expertise**
1 Initial	Recognition	Sporadic communication on the issues	*Ad hoc* approaches to process and practice			
2 Repeatable	Awareness	Communication on the overall issue and need	Similar/common processes emerge; largely intuitive	Common tools emerging	Inconsistent monitoring in isolated areas	
3 Defined	Understand need to act	Informal training supports individual initiative	Existing practices defined, standardised and documented; sharing of the better practices	Currently available techniques used; minimum practices enforced; tool set standardised	Inconsistent monitoring globally; measurement processes emerge; IT balanced scorecard ideas being adopted; occasional intuitive application of root cause analysis	Involvement of IT specialists
4 Managed	Understand full requirements	Formal training supports a managed program	Process ownership and responsibilities assigned; process sound and complete; internal best practices applied	Mature techniques applied; standard tools enforced; limited, tactical use of technology exists	IT balanced scorecards implemented in some areas with exceptions and noted by management; root cause analysis standardised	Involvement of all internal domain experts
5 Optimized	Advanced forward-looking understanding	Training and communication support external best practices and use of leading-edge concepts/techniques	Best external practices applied	Sophisticated techniques deployed; extensive, optimized use of technology exists	Global application of IT balance scorecard and exceptions noted by management globally and consistently; root cause analysis consistently applied	Use of external experts and industry leaders for guidance

4. Appendix D: Outcome / Solution Matrix

Appendix D: Outcome / Solution Matrix	Tools and Applications											
Business Outcome Requirement Needs =>	**CRM**	**Data Bridge**	**Community CRM**	**CMS**	**Doc Mgmt**	**Impact**	**List Mgmt**	**DAF Portal**	**Online Grants**	**Seamless Invest**	**Fund Accounting**	**Q & A**
A: CRM Installation												**B+**
B: CRM Everywhere												**B**
C: Web Presence												**A**
D: Document Mgmt												**B+**
E: e-Marketing												**A**
F: Self service portal.												**B**
G: Online grant request												**B+**
H: Online donation												**A**
I: Grant Application												**B**
J: Grant payment review												**D**
K: Grant Conditions												**D**
L: Workflow sequencing												**B-**
M: Event management.												**A**
N: Community Building												**C**
O: Outcomes measurement												**C**
P: Seamless Investments												**A**

Note #1: Shadings — Solid indicates a required tool. Shaded indicates that tool will enhance the solution.

Note #2: Q & A ratings:
- "A" – shrink wrap solution available
- "B" – tool available: requires configuration
- "C" – prototypes available: requires development
- "D" – requirements known, lacks prototype