The Integrated Feasibility Experiment (IFE) Process

J. Allen Sears
Corporation for National Research
Initiatives
1895 Preston White Drive
Reston, Va. 20191
asears@cnri.reston.va.us

Stephen E. Cross
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890
sc@sei.cmu.edu

ABSTRACT

In this paper, we describe a process used for guiding the evaluation and transformation process for language processing research and development. The Integrated Feasibility Experiment process is explained by describing the key six steps, and then providing a specific example to help understand how to implement the steps.

1. INTRODUCTION

The objective of this paper is to describe a reliable and repeatable process used to guide the development of information systems where technology teams must come together to implement a concept. This paper describes an "IFE process" that has been used successfully multiple times over the last eight years, and has served as both a framework for language experimentation, and as a vehicle for integrating and applying language technology components.

2. DESCRIBING THE IFE SIX KEY STEPS

The IFE process consists of six steps that guide development and experimentation. The emphasis placed on each step depends on the maturity of the technology and the involvement of the users. The six steps are as follows (note that the six steps are summarized in Figure 1)

2.1 Step #1: Scenario

Describe a scenario for employing the information technology that will allow everyone to visualize how the technology is to be used during a real situation. This step places emphasis on making the technology look and behave like a system. This is a critical step for two main reasons:

a. For the technology teams that are to integrate technology, the scenario provides a real and accessible description of how the technology should be used. This assists the teams directly in describing the architecture and components needed to build an information system for the given scenario.

b. The scenario is key in describing the intent of the information system to the operational user. Typically, operational users become involved in this scenario building process to give early and helpful feedback to the technology development teams.

2.2 Step #2: Architecture

Many people believe that describing the architecture is the key step in building and information system. However, if the ideas about components and interconnections are vague or incomplete, then the architecture step is actually best developed using a hypothesis and test process. In all cases the architecture must allow plug-and-play concepts that support the inclusion and reuse of mature processing components, plus the inclusion of new components that will be the focus of experimentation.

2.3 Step #3: Reuse Components:

The third step is to identify and make plans to reuse components that one will depend on during the IFE. This step is critical for the technology teams because many of the components to be used come from years of development and experimentation. With mature components populating a large share of the architecture, the development teams are then free to experiment with new components that are considered to be necessary for end-to-end processing. Moreover, the developers can experiment with data flow and interconnection strategies. This experimentation step is critical in order to transform into tomorrows' network-centric processing models supported by communication interoperability provide by TCP/IP processing.

2.4 Step #4: User Involvement

Obtaining operational user involvement early-on is an important step to support a technology transformation objective. The operational user will have insights and needs that cannot be predicted by the technology developers. Moreover, user involvement improves the interest, understanding and potential commitment to the technology. If user centered final exam metrics are stated clearly then they provide a useful objective to help focus technology development and implementation. This all may sound like motherhood, but it is a critical step that is missing often from technology development projects large and small.

2.5 Step #5: Rapid Prototyping

The use of a rapid prototype approach is not new. In the mid 1980s it became the key focus for specifying and building information systems. However, the rapid prototype process must be used in conjunction with other steps of the IFE, or else the development effort will end up as a simple demonstration that does not scale to real user needs. The spiral development model for development that emphasizes the “build a little, test a little” approach, should be used to keep development on track and headed toward the target needs of the user.

2.6 Step #6: Evaluation and Feedback

Metric-based evaluation is important for any development process. For an IFE the specification of usable metrics is not easy because the teams are coming together to build a “new” capability. The best approach comes by making an early commitment and following through with the measurement process and then later changing the evaluation process to better represent the emerging information processing capability. One should have measures for technology accomplishment and such measures should focus on component performance. In addition, one must have an overall “task performance” metric or metrics that reflect the needs of the operational user and the intent of the scenario.

Integrated Feasibility Experiment Steps

1. **Scenario** .. Helps to visualize the use of new technology
2. **Architecture** .. Components, interconnects, data flow, and processing model
3. **Reuse components** .. Must build on past accomplishments
4. **User** .. The user provides application pull, as opposed to technology push
5. **Rapid prototype** .. Build a little, test a little strategy to keep effort on track and on target
6. **Evaluation and feedback**: Metrics-based evaluations are key to understanding accomplishment

Figure 1: The six steps of an Integrated Feasibility Experiment

2.7 Historical Note

An Integrated Feasibility Development (IFD) process was first used in 1990 by Steve Cross and his team to guide development of the DARPA and Rome Labs replanning system called DART (Dynamic Adaptive Replanning Technology). DART was developed to assist logistics and transportation planners in scheduling the movement and deliver of people and materials. An operational prototype was actually used during the Persian Gulf conflict in 1990. The IFD name has been changed to IFE by replacing “Development” with “Experimentation” in order to emphasize the experimentation and scientific exploration aspect of the effort, but the steps of the process have remained the same.

3. WHY THE IFE PROCESS WORKS

Their three good reasons the process works and a forth explanation that deals with the basics of building and implementing information technology.

3.1 Application Pull

The scenario and user involvement (steps 1 and 4) work together to provide an “application pull” on the technology. To many efforts fail because they start with a new idea which is pushed and developed and then is found to be in search of (ISO) of a meaningful application. This “application push” model fails in most cases because no user is willing or able to invest in an acquisition follow-on process. Instead, the application pull process will address new information system introduction methods that take full advantage of commercially created information technology, and blend in radically new ideas that provide for scale and success. These steps insure transformation efforts will be based on innovation and speed.

3.2 Scalable Baseline

The architecture and reuse-of-components focus (steps 2 and 3) provides a baseline capability that will enable the information technology to scale up to deal with operational needs. Moreover, this investment in the software architecture provides the infrastructure needed to explore new ideas in an affordable and repeatable fashion.

3.3 Build A Little, Test A Little

Rapid prototyping and evaluation steps (steps 5 and 6) offer a simple and understandable approach to allow for incremental progress that is informed by failure as much as by achievement. This is key. Innovation must be allowed to fail just as long as the process moves forward and is informed in a positive way by the failure. Too many projects fail to provide for the process of managing risk and failure. Such projects are doomed to incremental advancement at best.

3.4 A Managed Process That Works

Some observers of the IFE process have said the six steps are necessary and sufficient to provide guidelines for information systems development and implementation. Necessary and sufficient does not guarantee success. It does however provide a small and simple set of steps that can help the technology community to shape information technology, and give it an outstanding shot at success. In most instances the IFE methodology has addressed crisis action and crisis response scenarios that address dynamic problems in the effective use of people, resources, information, and network-centric computing.

This methodology has been used to increase cooperation between defense and intelligence groups to develop command, control, computing, and intelligence infrastructure fundamental to developing new concepts of operation, and the foundation on which future capabilities are built.

4. AN EXAMPLE IFE

The following provides an example of the Strong Angel IFE used for the "PacTIDES" exercise in June 2000. This exercise was sponsored by the US Joint Military Command known as CinCPAC and included seven other nations and the United Nations. Both the accomplishments and the lessons learned will be covered. The Strong Angel IFE provided and outstanding framework for learning more about end-to-end language processing.

4.1 Strong Angel IFE Overview

Step 1. Scenario: The primary application focus for the IFE was the spread of disease, with special emphasis given to information processing techniques. The operational user was Dr. Eric Rasmussen, MD who was the Third Fleet Surgeon for the United States Navy. Dr. Rasmussen was most concerned about providing effective support and relief to people during "Humanitarian Assistance" operations that are becoming common through out the world. Examples like Bosnia and Kosovo come to mind immediately. The story line was that refugees were caught in a border location and world organizations were coming together to provide food, shelter, and security. The spread of disease soon became one of the top security risks. The TIDES system was used by the security teams to get timely information about relevant events so they could anticipate critical situations they may face instead of simply reacting to issues.

Step 2. Technology teams outlined a plug-and-play architecture called the "TIDES Portal" that was used to guide the development and experimentation process. The architecture was built on a client – server model where components for language processing were loosely confederated over the Internet.

Step 3: Component specification: The three primary information processing components were focused on detection, extraction, and user interaction. There also was a translingual component that provided two way translations to and from Korean. The scenario was expanded to include the treat of a missile launce from North Korea that could carry a biological war-head. The translingual component was an add-on rather than a main line processing component. There were seven different sources of news that was being processed to provide information to relief and security personnel. These sources included both text and speech information. The speech information was transformed into text and then became input to detection and extraction processing. The user interface component was the most difficult to construct because the underlying end-to-end processing model was emerging and changing each month. Moreover, the loosely coupled distributed processing model for the TIDES Portal was difficult to realize in a coherent user interface. This issue and other shortcomings are discussed in the lessons learned section of this paper.

Step 4: Operational User Involvement. The scenario definition process helped Dr Rasmussen and the other relief and security operators understand how the technology would come together to be used. The TIDES Portal and the "PacTIDES" experiments were use by representatives of several of the RIMPAC nations and also by United Nations personnel. For the first time ever the RIMPAC exercises conducted by seven nations: US, Canada, Japan, Chile, Australia, Korea, and UK, included a focus on a humanitarian assistance issues. For the first time users were able to understand in context the kinds of capability an automated information processing system such as TIDES may provide in the future. The potential for TIDES support received strong endorsement from these operators who are literally overwhelmed by data, documents, and email, but who are often starved for actionable information.

Step 5. The rapid prototype process was used to develop the IFE integrated system called the "TIDES Portal". Initial TIDES Portal implementation was tested in early 2000, and the final exam for TIDES Portal was conducted during Strong Angel was held In June 2000 on the Parker Ranch in Hawaii. The system was used by military and by UN World Food Program personnel. There was one situation where UN folks needed timely information about a situation in Africa, and the TIDES Portal came through. The UN team was impressed. However, most of the lessons learned at Strong Angel pointed to weaknesses in the TIDES Portal concept of operations. These weaknesses have become the main focus for development of IFE-Bio in 2001.

Step 6. Metric-based evaluation was used in Strong Angel with limited success. The weaknesses in the end-to-end processing capability of the TIDES Portal dominated the IFE and limited the ability of research groups to conduct full metrics-based evaluations in a meaningful way. This issue will receive more attention in during IFE-Bio final exams in June 2001.

5. LESSONS LEARNED

The Strong Angel IFE was judged to be a success even though several parts of the effort resulted in failure. The important point is that the TIDES Program learned from both the failures and the accomplishments and the lessons help guide the IFE process in 2001. The following provides an example of the Strong Angel IFE used for the "PacTIDES" exercise in June 2000. This exercise was sponsored by the US Joint Military Command known as CinCPAC and included seven other nations and the United Nations. Both the accomplishments and the lessons learned will be covered. The Strong Angel IFE provided and outstanding framework for learning more about end-to-end language processing.

5.1 Lessons Learned From Negative Examples in Strong Angel IFE

A. Process model was too uncoupled. Several groups came together integrated by only the Internet. The processing components were not synchronized and basically had little inter-dependency. Therefore, there was little in the way of information management within the infrastructure to hold the information processing model together.

B. Late-binding decisions about distributed processing burned up critical development cycles. The situation here is simple: initially the assumption was made that full Internet connectivity at T1 rates would be available, and then the assumption was changed to anticipate NO Internet connectivity outside of the camp. The change in the Internet connectivity and quality of service assumptions were made with two months to go. Most of the time was then spent on building local servers and processes that would simulate that external communications was in place. During the critical last two months critical development and testing was stopped and attention was turned to re-engineering the processing infrastructure.

C. Collection issues were not properly anticipated: The strengths of TIDES processing comes from end-to-end processing of streams of information from sources such as radio, TV, email, newswire, etc. Unfortunately the language detection and extraction communities are conditioned to processing from training and test sets provided to them in efforts such as TREC and MUC. Strong Angel concepts of operation actually required continuous processing of streaming information from multiple sources. These capture and processing priorities were not realized soon enough in the IFE process, and were therefore sorely lacking at the Strong Angel final exam.

D. TDT processing concepts were not included: The detection, extraction, and summarization process for Strong Angel anticipated Topic Detection and Tracking (TDT) capabilities, but the algorithms were never incorporated. This means that critical front-end filtering and grouping functions were missing.

5.2 Lessons Learned From Positive Examples in Strong Angel IFE

A. The Strong Angel team never imagined how difficult the living and information processing environment could be in a refugee camp. In fact there was fine grain dust everywhere and the power was intermittent. Better understanding of these environmental factors was a positive coming from the Strong Angel effort.

B. A key positive was developing the understanding for how detection, extraction, and summarization must work together with collection and distribution to provide an end-to-end processing infrastructure.

C. An important accomplishment occurred when UN folks wanted to know more about a growing crisis in Uganda after a humanitarian incident. It turns out that TIDES processing was able to give the UN contingent information that they needed that was current and multi-source. The UN folks were thrilled and amazed. Most amazing to the TIDES folks was the capability only used 10% of what was anticipated for TIDES processing. In other words, a very small and easy product provided significant value. There is great confidence that much more can be accomplished in IFE processing in 2001.

6. LOOKING AHEAD FOR THE IFE PROCESS

For TIDES two different but concurrent IFE processes are being pursued during 2001. First a team including MITRE, UMASS, NYU, and the Navy are developing IFE-Bio concerned with gathering real time information to aid in the analysis of spread of disease. A team of BBN, UMASS, and CIA are looking at automatically extracting information in real time from a wide range of Arabic open source material. When ready and mature, technology and language processing techniques will be incorporated into Foreign Broadcast Information Service (FBIS) processing. A short abstraction of IFE processing six steps is provided in figure 2 for the 2001 effort called IFE-Bio. In addition the DARPA Communicator is using the IFE process to help in the development and transformation process for dialogue interaction. The Communicator IFE process is being continued aggressively in 2001 by a team including Lockheed, MIT, and the United States Marines Corps. For DARPA Communicator the initial LCS Marine IFE process has matured and is now being applied to a wider range of military exercises. Valuable lessons learned emerge from every exercise and aggressive concepts of operation are being investigated.

IFE - Bio: Example for TIDES

- **Scenario:** TIDES technology will be used to extract information about spread of specific diseases. Crisis response teams will pose ad hoc questions to the system.
- **Architecture**: End-to-end processing to include source capture from audio and text, TDT processing, extraction, summarization, and finally alerting & distribution.
- **Reuse components**: IFE - Bio will use language processing components from NYU, UMASS, and MITRE
- **User**: LCDR. Eric Rassmussen, former Third Fleet Surgeon, will stress test the IFE crew to see how well they respond to questions that would come up during a crisis.
- **Rapid prototype**: Initial build for 27 Feb, then mid-term in April will test second build, finally the June 2001 will test the final build of the prototype.
- **Evaluation and feedback**: Technical evaluations will cover all key components. The user evaluation will focus on ease-of-use and performance improvement

Figure 2: Overview example of IFE-Bio