The Future of CAE Software
How to Achieve Process Automation

Brian Huf
Ford Motor Company

Contents:

1.0 Abstract:

Achieving Process Automation is goal of all software – Automatically do what a user needs to get done. CAE software is no exception. In previous years, technological advances have driven most of the progress in CAE software. CAE software development is now starting to follow the lead of many other software companies by leveraging the concept of Process Guidance (a.k.a. Wizards). This approach has been found to be very effective as well being an advantageous resting spot on the road to Process Automation. Developing and deploying Process Guidance software introduces new questions and opens new possibilities. Several scenarios and strategies are discussed and analyzed. Also included is a list of criteria that could be used to rate prospective software companies on their ability to successfully lead Ford on the path to Process Automation via Process Guidance.

This paper proposes that Ford:

- **Pursue Process Guidance software development for <u>all</u> CAE processes**

2.0 Separating Process and Technology:

The concept of process automation has been around for quite some time. It's natural to want to automate everything and aggressively pursue Process Automation. The concept of Process Guidance has recently gained attention. Process Guidance basically accepts the fact that it is not possible to completely automate everything (unlike Process Automation). Process Guidance guides a user through a process, like the way Wizards helps us in TurboTax®.

2.1 The Difference Between Tools and Process Automation:

The **differences between tools and process automation are usually blurred.** It can be difficult to discern one from the other. Consider this analogy over time:

No.	Manufacturing Analogy	CAE Mesh Creation
1	Rocks & Stones	Text Editor
2	Hammer & Anvil	Preprocessors & Mapped Meshing
3	Stamping Machines	Preprocessors & Automeshing
4	Rapid Prototyping	Batch Meshing

As time progresses, tools become more powerful and general. The amount of time to perform the same task greatly diminishes while complexity increases.

Process automation needs to be taken into the context of the current time. Back when the text editor was the tool for creating FE models, it would be entirely reasonable to consider a Preprocessor as a process automation tool. A preprocessor automatically writes all the necessary text to describe an FE model. In light of this, the distinction between Process Automation and Tools is rather blurry. One could argue all new tools automate processes that were previously done manually; therefore all tools are process automation tools.

Consider this simplified process of meshing. A new user unfamiliar with FE modeling would probably find a process similar to this on the web or in a manual. The first step, reading CAD, would probably be described in more detail. The documentation would demonstrate 'click-by-click' how to import a CAD file.

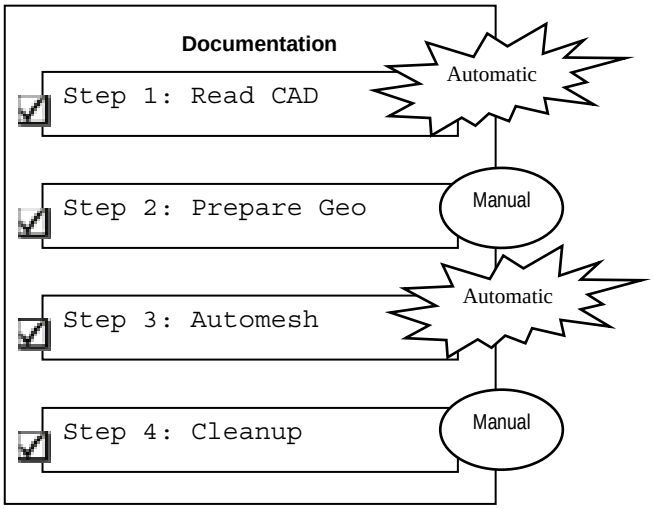

Figure 1 – Manual Process

Now consider Batch Meshing – by definition, a completely automatic process. Transforming a manual procedure into an automated process usually requires development in two separate areas. First is technology-oriented development and second being process-oriented development.

1. **Technology-Oriented Development relates to the hard-core algorithm development** necessary to automate certain tasks. For Example, in batch meshing - geometry preparation stage requires the most Technology Development. At this moment, only a trained person is able to do this step successfully. It has been very challenging to develop an algorithm to do what a person can readily do. It is during technology development when a significant amount of time and energy is spent. Emphasis here is usually on sophisticated logic and algorithm development.

2. **Process-Oriented Development is the glue**, which connects sub-tasks together. Typically, process-oriented development is rather straightforward and can be done quickly and inexpensively. Emphasis here is on logistic and communication development.

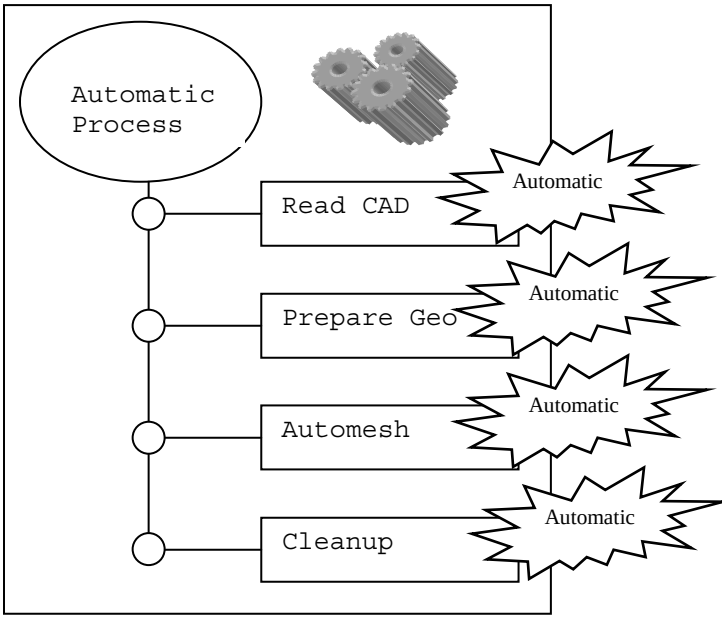

Figure 2 – Process Automation

In many cases, Process Development and Technology Development are done at the same time, and the code is intertwined. Development of this nature is undesired as it conflicts with a basic principle of software development – modularity. As will be shown later (Section 4.1), embracing modularity and open-architecture creates new possibilities and many advantages.

It should be noticed that for Process Automation to exist, every sub-task must be automate-able. All it takes is one manual step and the house of cards would seem to fall. It can be tempting make a sub-task automated before it is ready. Although Process Automation was achieve, quality and robustness could be at risk. This is a common example of the balance between speed (Automation) and robustness. Luckily, a solution to this dilemma can found through by applying the concept of process automation. A solution, in fact, that appears to offer the best of both worlds.

2.3 Process Guidance Software

Consider applying the concept of Process Guidance to the meshing process. While Process Automation needs all the sub-tasks automated, Process Guidance is able to use the sub-tasks, as is (automated or manual). A "wizard" is like the conductor of an orchestra. It doesn't do any of the work per se, but is does guide the user through all the sub-tasks.

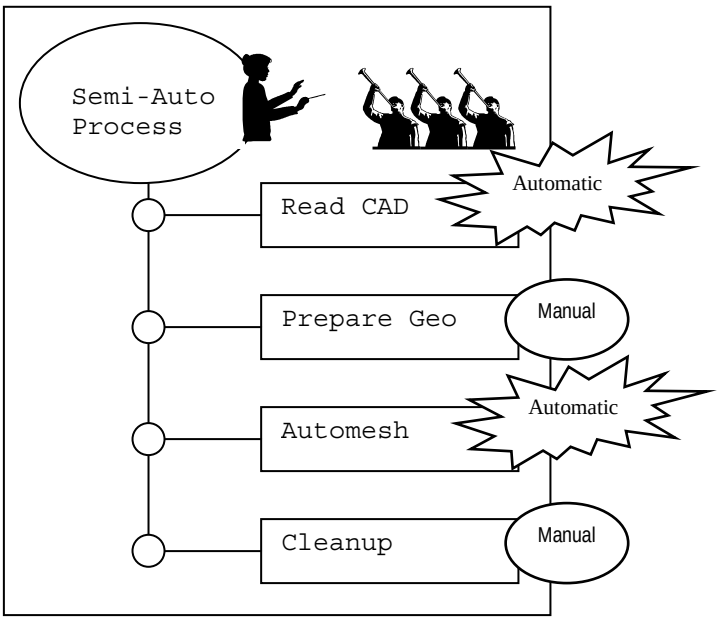

Figure 3 – Process Guidance

The first Process Guidance software in CAE I'm aware of was a Static Bending/Torsion Analysis tool by EASi Engineering. The tool would guide a user through preparing an analysis. Once analysis results were calculated, the tool would post process the raw data and automatically fill in a report.

The fundamental difference with this software was the acceptance that the process was not completely automate-able. For example, the technology does not exist where a tool can automatically locate a shock tower. When this happens, the Process Guidance software asks the user to locate them. Once located, rigid elements and boundary conditions will be automatically created.

One clever feature of the software is to save a session file. If a new model is "sufficiently" similar to a previous model, it is then possible to automatically select the shock towers and process in a highly automated fashion. This isn't very high-tech solution, but it is quite effective.

The traditional train of thought would not allow the semi-automated nature of this tool. In the past, development would focus on a tool that automatically identifies this shock tower. Once this tool exists, the whole process would be completely automated. This would seem to be the ideal course of action. The flaw is this: developing a shock tower finder would be extremely difficult at this time. In the past, development would have two choices; spend a significant amount of money (and time) to create a shock tower finder or not pursue the Static Bending/Torsion tool.

Luckily, someone applied the rather common concept of Wizards to CAE software and a whole new approach can be pursued.

2.3 The Different Levels of Automation

Process guidance can be seen to be an intermediate spot on the road to automation. There are different levels of sophistication when it comes to processes.

Level 1: Process Documentation:
This is seen as ground zero. Documenting processes on the web or any other media would be to always be the first step. To utilize, users are expected refer to certain documents and then read, interpret, and execute their procedures. It is here where most of Ford operates. CTEP's are the most obvious example of Process Documentation.

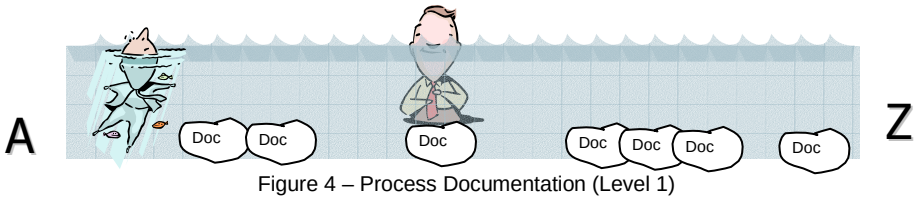

Figure 4 – Process Documentation (Level 1)

Level 2: Process Guidance:
Software guidance is a way of guiding users through a process while automating everything possible. It is quite reasonable to re-package all process documentation, put it into a single environment, make loose hooks to technology, and call it Process Guidance. The most obvious example of process guidance is TurboTax®. TurboTax© does most of the calculations and logic, but when user input is needed, the software will ask for it.

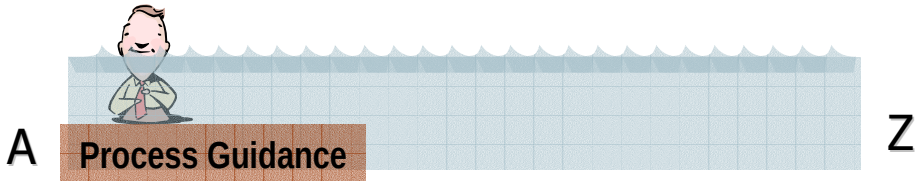

Figure 5 – Process Guidance (Level 2)

Level 3: Process Automation:
Push a button and the software does everything. Example: ActiveSync synchs the calendar on a PocketPC with a desktop computer. In most cases, ActiveSync is able to synch the two machines with no user intervention.

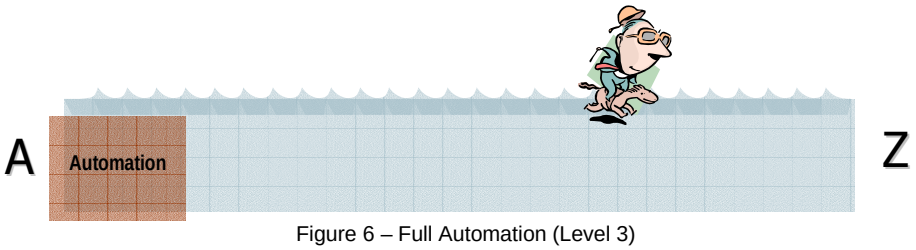

Figure 6 – Full Automation (Level 3)

3.0 Advantages of Process Guidance:

Process guidance would seem to present the best of both worlds. It provides many of the benefits of Process Automation, but without the expense and long lead.

Benefits of Process Guidance software compared to Process Documentation (current):

1. Improved Quality
2. Improved Speed
3. Low Cost and Timely Development
4. Stepping-stone to Process Automation

3.1 Improved Quality

Process guidance improves quality by ensuring users follow a common corporate procedure. This results in consistency (e.g. model consistency, result consistency, etc.), which is something not achieved today.

Currently, most of Ford's procedures and best practices reside on the web. Although this is a good start, web documentation is rarely used. There is no assurance users are using the current procedures. In fact, most users will learn a procedure once and not refer to it again. New procedures have always been challenging to deploy.

With process guidance software, all users would follow the current corporate procedure. Process guidance, in effect, encourages user to do the right thing. Not following the current procedure would actually be more difficult to do and take more time – something users probably wouldn't do.

3.2 Improved Speed

Process guidance improves speed by supplying users with the most efficient procedures. Process guidance also allows further automation due to the nature of integrating process information into a tool.

Current tools need to be extremely general due to their wide range of uses. With this generality comes complexity. Complexity leads to longer learning curves and a slower, less efficient system. Due to the nature of the process guidance, these general tools will now be able to take advantage of knowing the process. This will certainly lead to automation without technology development.

For example, in the static bending/torsion tool, a rigid element will need to be created at each of the shock towers. Since the user already identified the shock towers, the software could automatically locate the holes, create the rigid, and apply the necessary boundary conditions.

3.3 Process Guidance is a Low Cost Solution

Process guidance software can be developed without significant technology developments.

Unlike process automation, process guidance does not require all sub-tasks to be automated. Achieving process automation usually requires a significant amount of R&D, which quite expensive and time consuming.

3.4 Process Guidance is a Stepping-stone to Process Automation

Process Guidance is a natural stepping-stone to Process Automation. Once Process Guidance is in place, developers will then focus their attention on the sub-tasks and make them automated. When all sub-tasks are automated, Process Automation is achieved naturally.

4.0 Software Development Strategy:

What is the best path to Process Automation? Imagine a process from A to Z.

Strategy #1: Identify Bottleneck and Fix

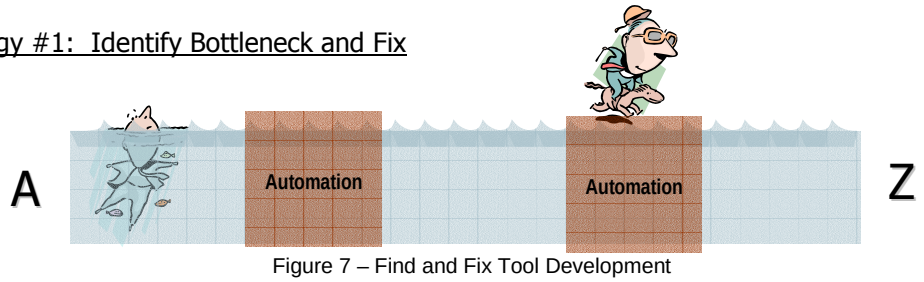

Figure 7 – Find and Fix Tool Development

Even though a few tools are in place, the water is over the user's head much of the time.

The find and fix has been the dominant strategy in previous years. Various bottlenecks would be identified in a process and software development would then create a tool (technology) that would attempt to completely automate this process, thereby eliminating the bottleneck.

Advantages:

- Bottlenecks are completely eliminated and measurable improvements are noticed
- Straightforward approach to achieve complete automation for all processes
- Process Automation can be achieved over time

Disadvantages:

- Automation usually requires technological development which can be expensive and time consuming
- Sometimes difficult to know where the bottlenecks reside

Example:

Meshing has been a bottleneck in CAE in past years. Software development has focused on improving this process and has achieved some success. Due to the challenging nature of this process, complete automation has yet to be achieved. Batch Meshing (complete automation) has been elusive and it is actively being developed.

Just two years ago it was not unreasonable to spend a day meshing a single part from CAD. Today (without Batch Meshing) people are able to mesh upwards of 30 parts a day. Even though meshing has stopped being the bottleneck, meshing development projects have gained such momentum that complete automation is the only measure of success. As meshing has improved over time, other processes have become more time consuming.

Strategy #2: Implement Process Guidance, then Develop Technology

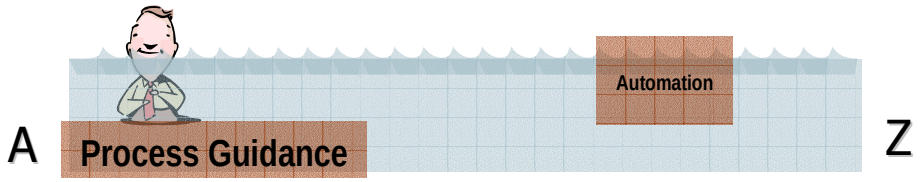

Figure 8 – Adding Automation to a Process Guidance Wizard

Even without automation, Process Guidance provides significant value over simple Process Documentation. In addition, an infrastructure is created that facilitates the implementation of new technology as it becomes available.

4.1 How Tightly Should Process and Technology be Integrated?

It is valuable to recognize that the level Process Guidance is a continuum. Many different levels of integration exist. Likewise, the distinction between what is Process and what is Technology can be blurred as was discussed in 2.1. Nevertheless, it is valuable to make the distinction, and then ask how tightly they should be integrated.

A simple analogy is an orchestra. The conductor resembles the Process-Oriented software and the instruments resemble technology.

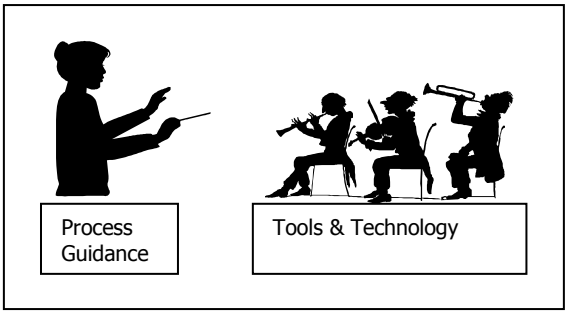

Figure 8 – Orchestra Analogy

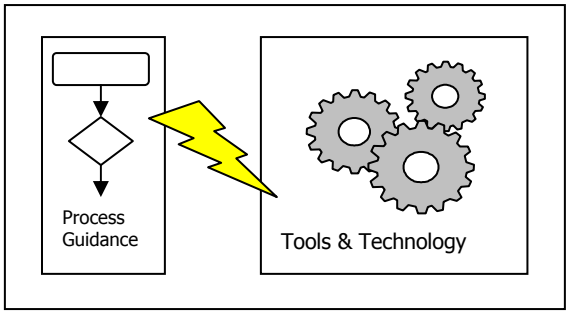

Figure 9 – Separating Process from Technology

The critical thing to notice in Figure 9 is the Lightning bolt. This represents the communication between the process-oriented software the technology-oriented software. Until recently, software development was such that integration of the two required the same software company. With the nature of plug-n-play pervasive throughout software development, it is now possible to develop a software environment, whereby the pieces are developed separately and possible in separate companies.

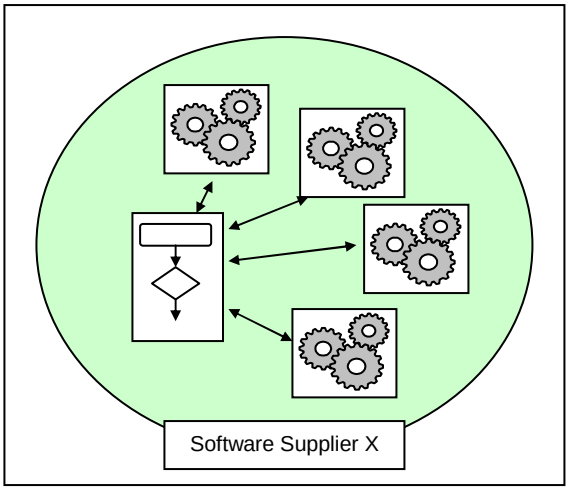

Figure 10 – Tight Integration (Little Flexibility)

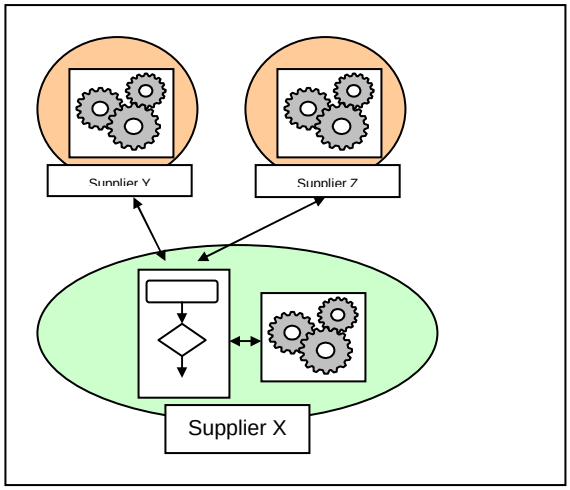

Figure 11 – Loose Integration (Maximum Flexibility)

Figure 11 suggests that it is possible to utilize the best technologies and leverage them in a single environment. Modularity across software suppliers has only just recently become reasonable to expect. Most CAE softwares are moving toward an open architure allowing the functionality to be used by other softwares.

Currently, one critical restriction must be recognized. **Only automated processes can loosely integrated.** For example, suppose Supplier Y has the best "painting" tools. However, if "painting" is a rather interactive process, it is not reasonable to expect Supplier X would be able to integrate the painting functionality. Recognition of this restriction should be factored in during the supplier selection process.

It should also be noted that this restriction only represents the current moment. Software technology is currently available whereby loose integration and interactivity can coexist. This mechanism is demonstrated in the ability to edit a PowerPoint® object within MS Word®. The restriction exists because of UNIX and a slight lag between CAE software development and main stream software development.

Discussion: Tight Integration vs. Loose Integration

Advantages of Tight Integration:

- Less Expensive. A single supplier can probably do the job cheaper than multiple suppliers.
- Less Ford Supervision. Loose Integration will probably require play the role of integrator between various software suppliers.

Advantages of Loose Integration:

- Superior Functionality by using "most" of the best-in-class technology.
- Upgrade-able. Future improvements in technology can be "plugged" into an environment with loose integration. With Tight Integration, suppose Supplier X is no longer the best tool for painting. Tight integration would not allow integration of the new, best-in-class, technology.
- Adaptability. Loose integration supports competition and does not necessitate reliance upon a single supplier.

Development time seems to stand on equal ground. Tight Integration should be considered as less complex, however loose integration could take advantage of having multiple software suppliers working in parallel.

	Tight Integration	Loose Integration
Initial Cost	Advantage	
Development Time	Equal	Equal
Productivity/Performance		Advantage+
Flexibility		Advantage+

Based on this analysis, the advantages of Loose Integration are worth the extra cost.

4.2 How Exactly Does the Process Software "Hook" into the Technology Software?

Like everything else, the level of integration between process and technology is a continuum. One simple way to describe this continuum is to describe its extremes.

Definition of Process Hook: Clicking in the "process window" causes action in the "tool window". The process window and the tool window are "hooked" together. Note that two separate windows are not completely necessary, however, it is envisaged that two will somehow be separated.

No Process Hooks: A process window would be on the screen **describing** what to do in the tool window

Really Good Process Hooks: The process window is able to drive the tool window to such an extent, that the tool window no longer needs to have a GUI (no menus).

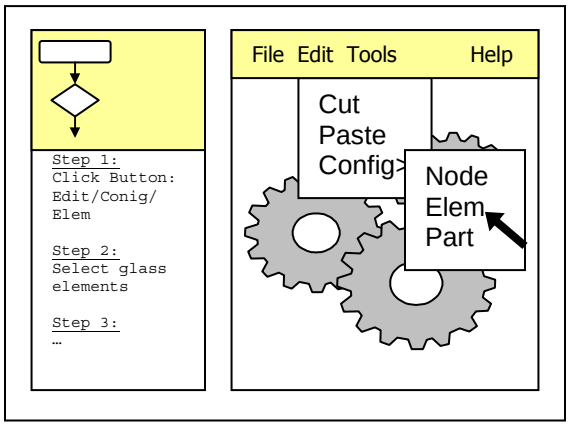

Figure 11 – No Process Hooks

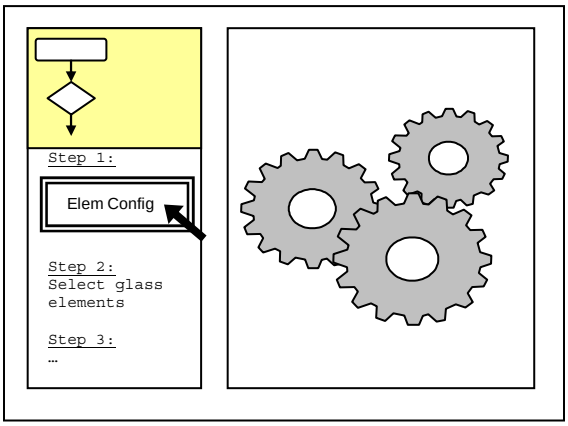

Figure 12 – Really Good Process Hooks

Figure 11 could be considered as Process Documentation repackaged in such a way that a user would use the procedure and tool simultaneously.

4.3 How to Progress from Level 1 to Level 2

It is proposed that the following steps are used when transitioning from Process Documentation to Process Guidance.

1. Ensure Process Documentation is complete.
   - It is important that process be complete and well understood. Any process gaps will be highlighted quickly when moving on to step 2.

2. Re-package documentation into a single environment.
   - This is most easily done by making web pages that are well connected (hypertext links). It important that process dependencies be documented as well as process in/out relationships.

3. Develop process hooks
   - The necessary level of process hooks is at this point, unclear. It is suggested a relatively low level of hooks be provided and then make a reassessment.

4.4 How to Progress from Level 2 to Level 3

Once a minimal level of Process Guidance for an entire process is in place, it is then appropriate to work towards Process Automation.

This can be seen as an iterative process by making incremental improvements. Improvements can be classified into two different categories. Which one choose should be based on a business case by business case basis.

1. Process Hook Improvement
   - Improving the process guidance hooks will undoubtedly improve the efficiency of the tool.

2. Technology Improvement
   - Identify a technology improvement with the best cost/benefit ratio and implement into the process guidance environment.

After enough iterations, complete Process Automation will be achieved.

5.0 Software Supplier Partnership Strategy

5.1 How to Evaluate a CAE Software Supplier?

Criteria to evaluate a Software Supplier (1-10, Little Capability – Mastery):
1. Technical Depth - Technical Experience & Expertise
2. Process Depth – Process Oriented Experience & Expertise
3. Breadth – Multi-Attribute Experience
4. Reliability – Have past projects been successful?
5. Creativity - Surprise & Delight Factor
6. Foundation – Can they re-use 90% of what they already have?
7. Speed – Lines of code per day
8. Support & Training – Can they support us?

5.2 How should Ford be involved in development of Process Guidance Software

Certainly, Ford is not able to do all the development. In fact, one software supplier may not be enough (Section 5.3). This section explores the question, **"Who does what and how much of it?"** The main driver for answering this question is: **Who has the capability and talent?** For example, when it comes to coding, software suppliers may have an advantage; therefore it would be best to have them do the coding.

Possible Scenarios:

Scenario #1: Ford to deliver software vision and goals

This would be the equivalent of giving a software supplier this paper and asking them to develop the process & specs and begin development. Ford would oversee the project over time, but would not "micro-manage" the project.

Key Assumption: Software suppliers are best able to create CAE processes (given basic requirements) and translate them into software specifications

Advantages: With creative freedom, the software supplier may be able to surprise and delight Ford.

Scenario #2: Ford to deliver process documentation

Ford would provide the software supplier complete Process Documentation. The supplier would then determine the best way to achieve process guidance/automation.

Key Assumption: Ford understands CAE processes best. Software suppliers are best able to create software specifications.

Advantages: Ford still uses Common Best Practices and software suppliers have the creative freedom with implementation.

Scenario #3: Ford to deliver software specifications

Ford would provide the software supplier detailed specifications on the touch and feel of the software to be developed. The software supplier would follow the specs precisely and deliver exactly what is expected.

Key Assumption: Ford best understands how the software needs to behave. Software suppliers are best able to do the software development.

Advantages: "No surprises". Once specs are created, development time is fast.

Scenario #4: Process development at Ford ("Super User")

Ford would use one of the readily available "Process Helpers" (Altair, Sofy, and EASi all have one) to aid in Process Development. The software supplier would focus on technology and ensuring the proper interfaces to technology are available.

Key Assumptions: Process Development is a highly iterative process with many unknowns.

Advantage: Ford "owns" the process software and can readily updated over time. Extra attention can be paid to the pug-n-play aspect of technology.

5.3 How Many Software Suppliers to do Process-Oriented Development?

The main driver here is: **Is a software supplier able to development process software that guides a user through another supplier's technology software?**

Scenario #1: Choose a single third party software to do Process Development

It may be possible to have a non-CAE software company develop the Process-Oriented software necessary for Process Guidance.

Key Assumption: Third party has enough expertise to be successful.

Advantages: Improved integration. The third party supplier would not compete with the technologies it's integrating. A single, process guidance interface.

Scenario #2: Choose a single CAE software supplier to do Process Development

Ford would choose the most capable process oriented CAE software supplier.

Key Assumption: A CAE software supplier is able to successfully integrate another supplier's (possibly competing) technology. A single, process guidance interface.

Advantages: CAE expertise in the development of process-oriented software.

Scenario #3: Choose a multiple CAE software suppliers to do Process Development

On a process-by-process basis, Ford would choose the CAE supplier that has the most "mission critical" technology.

Key Assumptions: The supplier doing Process-Oriented development will not be able to integrate another supplier's (possibly competing) technology. Multiple process guidance software is ok.

Advantages: Maximum integration between process and technology.