



# CAPITAL LETTER

## In this Issue

Using a Hierarchical Process Design.....1  
 From the President.....2  
 From the Editor.....3  
 Chapter Chatter.....4  
 Membership News.....4  
 Congratulations to our Chapter Members.....4  
 Member News.....4  
 Upcoming Chapter Events.....5  
 Newsletter Events.....6  
 You Shop Amazon, We All Win!.....6

### Quotable Quote:

Quality is never an accident; it is always the result of high intention, sincere effort, intelligent direction and skillful execution; it represents the wise choice of many alternatives.

**William A. Foster**

## Using a Hierarchical Process Design

By Basil White

This article presents an approach for describing a business process and identifying the potential failure points. By using this approach, usability practitioners can collect consistent data for multiple usability tests, and also compare those data to identify patterns of success and failure. The Hierarchical Process Design (HPD) achieves the following:

- Creates a taxonomic model of an IT product
- Directs the structure of all product communication
- Directs business cases
- Directs product design
- Directs product implementation
- "Cuts the grapevine" of unnecessary reinterpretations from business goals to design specifications to products to user instructions

- Scales beyond the product it represents to larger product suites and organizational segments
- Helps program managers to represent their IT suite as the automation of an organization's business objectives
- Provides a framework for product usability assessments by defining the goals of an IT product in a single representation throughout conception, design, development and implementation
- Scalable to accommodate changes in program scope and resources

HPD activities can span the entire lifecycle of a product. This article focuses on how to use HPD to:

*[Continued on page 7](#)*

## From the President

Creating and supporting a forum for communities of practice in the profession of technical communication

**Society for Technical Communication ( STC)**

901 North Stuart Street,  
Suite 904

Arlington, VA 22203-1822

Phone 703-522-4114

<http://www.stc.org>

**Region 2**

<http://www.stcregion.org/region2/index.shtml>

**Washington, DC Chapter**

P.O. Box 14125

Silver Spring, MD 20911

<http://www.stcwc.org/>

**STC Washington, DC**

**Chapter Capital Letter**

**Managing Editor**

Kathy Bell

[news@stcwc.org](mailto:news@stcwc.org)

**Production Manager**

Jennifer Reed

[JLReed4@gmail.com](mailto:JLReed4@gmail.com)

**Copy Editor**

David Dick

[DavidJDick2000@yahoo.com](mailto:DavidJDick2000@yahoo.com)

The *Capital Letter* is published six times per year by the Washington, DC Chapter of the Society for Technical Communication. Its purpose is to provide articles and essays in the field of technical communication as well as inform chapter members about special functions and regular activities of the chapter and Society. Material for publication should be submitted to the editor:

**Kathy Bell at**

[news@stcwc.org](mailto:news@stcwc.org)

This newsletter invites writers to submit articles that they wish to be considered for publication.

**Kathy Bine**

**Recognize Your Excellence**



It's important to keep your objectives in mind, whether you are creating flow

charts, writing procedures, or climbing a mountain. It would seem that the objective of climbing a mountain would be obvious: reaching the summit. Leadership Day at the STC Annual Conference featured a keynote message by Rob Ziegler. Rob is chief operating officer at IHI Environmental. His passion, however, is mountain climbing.

His most compelling story was of climbing a mountain in Pakistan with Adrian Burgess, a very experienced climber. Adrian prioritizes the three objectives of mountain climbing:

- 1) Be safe
- 2) Maintain friendships
- 3) Summit the mountain

"Thanks for helping me get my first summit in Pakistan," Adrian said to Rob when they reached the summit. Rob didn't quite know what to say, since he knew this climber had been to mountains in Pakistan 22 times; this doesn't include the other mountains he's climbed. I remember Rob's face as he showed a slide of three people standing next to a van. "Two of these guys died shortly after this photo was

taken," he said. Approximately one out of ten mountain climbers die while climbing, in part because many climbing teams place a higher priority on summiting the mountain than on being safe or maintaining friendships. Adrian is alive because he would rather stay alive than summit the mountain under less-than-safe conditions. It was clear to me that Rob sees Adrian as a role model for how to survive mountain climbing.

Adrian would also rather ensure that he still has friends when he comes off the mountain, since he will probably be back. When Rob and Adrian returned to camp, the homebrewed beer that Adrian had started in two blue plastic drums was ready to drink. When the team of porters, cooks, and other support personnel had completed their tasks, Adrian paid each of them and thanked them personally. "We couldn't summit without these guys," said Rob. You have to recognize the people who carry your water, who make your objectives possible, but who you can so easily overlook. That's what maintaining friendships means: taking time to recognize what those around

*Continued on page 5*

## From the Editor

Kathy Bell

We have a special feature this month. When Basil White offered us the opportunity to publish his article in our newsletter, we felt it was a chance to explore an important topic in our field in more depth. Thank you Basil White for allowing us to publish a local, informed view on usability. This edition of our newsletter wraps up our STC season. We do publish a summer newsletter, but it is more focused on the upcoming fall season. I wanted to take a moment to thank those people that have really pushed our newsletter forward this year. Thank you to Pam Daily for her editing skills, her expertise, her advice, and willingness to answer what I'm sure felt like a million questions. David Dick is always willing to help out in any way that is needed. I appreciate his copyediting skills, his ability to always come up with an article, and his expertise at knowing how to create a better newsletter. Thank you to Bobbie Dofflemyer for her layout

expertise and willingness to help Jennifer when it was time to pass this job along. Jennifer Reed offered to take over the layout position and agreed to iron out producing the newsletter in a new format. Thank you. Kristen Sweet helped me numerous times to find someone willing to write an article for an event. Many members during the year offered to write articles on different events. These articles have given our newsletter its local voice. It was great to have so many people to count on, often in a pinch. Thank you to Kathy Bine for her encouragement, patience and support. She was always the calm voice at the other end of the telephone. Let us know what you think about exploring different topics at a more in-depth level. Have a great summer!

### *We Want Letters to the Editor*

Do you have something to say about a recently published article? Was there incorrect information we printed that you want to point out? Do you have an idea that you would like to share that may make the newsletter be a more useful and enjoyable resource? Contact us at [news@stcfdc.org](mailto:news@stcfdc.org).

### Chapter Leadership President

Kathy Bine

[president@stcfdc.org](mailto:president@stcfdc.org)

### First Vice President

Melissa Brown

[firstvp@stcfdc.org](mailto:firstvp@stcfdc.org)

### Second Vice President

Kristen Sweet

[secondvp@stcfdc.org](mailto:secondvp@stcfdc.org)

### Secretary

Eddie VanArsdall

[secretary@stcfdc.org](mailto:secretary@stcfdc.org)

### Treasurer

E.C. Eklund

[treasurer@stcfdc.org](mailto:treasurer@stcfdc.org)

### Immediate Past President

Daree Allen-Woodard

[pastpres@stcfdc.org](mailto:pastpres@stcfdc.org)

### Region 2 Director-Sponsor

Vici Koster-Lenhardt

[ds2@stc.org](mailto:ds2@stc.org)

### Committee Managers

#### Competitions

Melissa Brown

[competitions@stcfdc.org](mailto:competitions@stcfdc.org)

#### Employment

Ann Ray

[employment@stcfdc.org](mailto:employment@stcfdc.org)

#### Events/Knowledge

##### University

Kristen Sweet

[events@stcfdc.org](mailto:events@stcfdc.org)

##### History

Harold Hofstad

[history@stcfdc.org](mailto:history@stcfdc.org)

##### Membership

Laura Jackel

[membership@stcfdc.org](mailto:membership@stcfdc.org)

##### Newsletter

Kathy Bell

[news@stcfdc.org](mailto:news@stcfdc.org)

##### Nominations

Mike Cleary

[MCleary216@aol.com](mailto:MCleary216@aol.com)

##### Speaker's

##### Bureau/Webmaster

Cynthia Lockley

[webmaster@stcfdc.org](mailto:webmaster@stcfdc.org)

##### Strategic Planning

Janet F. Scharp

[jscharp@aim4value.com](mailto:jscharp@aim4value.com)

# Chapter Chatter

## Membership News

### New Members Who Joined Between February 16, 2005 and May 15, 2005

Denise Axtell	Bonita Dooms	CJ Huether	Alice Nielsen
Andrew Baskin	Adrienne Elefantis	Christina Huth	Patricia Palmer
Judy Biedrycki	James Emery	Candace Jenkins	Renoir Pope
Kathleen Bouchaud	Holly Fallat	Laura Kidd	Beth Rabinowitz
Samuel Brutcher Jr.	Bronwyn Gagne	Cara Kirchoefer	Cynthia Rezetko
Ginny Cathcart	Christine Gard	Aphrodite Knoop	Alesya Semukha
Yolanda Robinson	Sobhna Garg	Kevin Komatz	Robert Thayer
Darricarrere	Carol Gillis	Ellen Loeb	Nancy Vogel
Gregory Deahl	James Hodson	Robert Moody	Evan Weber
			Erin Willis

**Total WDC Members: 513  
Welcome to STC!**

## Congratulations to our Chapter Members

### *Distinguished Service Award*

- Carolyn Klinger** – Chapter Service Award
- Allen W. Rotz** – Usability SIG Service Award
- Cynthia A. Lockley** – AccessAbility SIG Service Award

### *International Competition*

**Capital News** – Excellence in the Newsletter competition and Most Improved for chapters with membership counts above 600

### *Members whose articles appear in the June issue of InterCom*

- Kathy Bine and David Dick** – “Introduction to Documentation Management for Lone Writers”
- Derek Torres** – “Third-Party Contracting as a Lone Writer”

## Member News



We welcome David Dick who made his move from Brussels to Virginia on May 11. David is co-founder and former president of the Belgium Chapter, editor of *Usability Interface*, winner of the

Usability SIG Distinguished Service Award, committee member to revise the annual STC newsletter competition rules, and volunteer of our Chapter newsletter.

Meet David at our Volunteer Recognition Luncheon.

## Upcoming Chapter Events



Event:	Volunteer Recognition Luncheon
When:	Saturday, June 4, 2005 11:00 a.m. – 1:30 p.m.
Where:	Capitol City Brewing Company 2 Massachusetts Avenue, NE Washington, DC 20002
Cost:	\$ 0 Current chapter, incoming officers, special guests of the chapter \$10 Current chapter volunteers, chapter speaker, corporate sponsors \$20 STC Members \$25 Nonmembers \$30 Walk-ins
RSVP:	Please RSVP by Wednesday, June 1, 2005. To reserve, complete the reservation form at <a href="http://www.stcfdc.org/prgm_resrv_jun05_volunteers.shtml">http://www.stcfdc.org/prgm_resrv_jun05_volunteers.shtml</a> .
Info:	During the chapter year, our members work hard to increase the value of Society membership by providing networking and learning opportunities, organizing competitions, communicating career information and job opportunities, raising money for scholarships, and many other activities. Come and help thank our volunteers and welcome our new leadership.

*[Continued from page 2](#)*

you are doing, and making sure that they know you appreciate them. It can also mean doing something more, something that may look crazy and unnecessary, like home brewing beer at 14,000 feet.

Rob hasn't summited every mountain he's attempted. Sometimes he has come close and had to turn back. Sometimes he's been with the advanced team that made it to the penultimate camp and helped those who summited descend safely. Sometimes the weather meant that the team never had a chance to make an attempt. This part of Rob's story I find most compelling. There's a grace in simply overcoming one's apprehensions and attempting something that's challenging and maybe dangerous. If you never try, you'll never know whether you might have succeeded.

I haven't accomplished the goals that I set for myself as president of STC WDC. I know at times tempers have been strained. It has been an amazing experience, though, and I'm grateful to you for the chance to serve as your chapter president.

At the Volunteers Recognition Luncheon on June 4, our chapter will recognize those who made so much possible for our members: the technical publications, technical art, and online competitions, the events, the newsletter, the postcards you receive about events, strategic planning, financial management, and all the other details that I haven't had to manage personally. I hope you can join us as we look over the path we have traveled and rest on our laurels.

---

## Newsletter Events

---

### Important Newsletter Deadlines

- All Content: 10<sup>th</sup> of even numbered months
- All Edits to Production Designer: 20<sup>th</sup> of even numbered months
- Final Review of Layout: 25<sup>th</sup> of even numbered months
- Publish Online and Send to Printer: 1<sup>st</sup> of odd numbered months

### Themes for Editorial Calendar 2004-2005 (six issues per year)

- September: Volunteerism
- November: Outsourcing
- January: The face of technical communications: A management perspective
- March: Job strategies
- May: Technology trends
- July: Lessons learned

---

## You Shop Amazon, We All Win!

---

Keep the STC Washington, DC chapter in mind when you make your purchases on Amazon.com. Links from <http://www.stcwdc.org> to Amazon.com contain our chapter's ID for the Amazon Associates program. Without adding a penny to your cost, the chapter earns a referral fee on items offered by Amazon or third parties. Gift certificates and items sold through auctions, zShops, and sites to which Amazon.com links, such as CarsDirect.com, are excluded from the referral bonus.

Since the chapter became a member of this program in April 2001, the chapter has earned over \$220. To boost our sales, we recently added a page of reference books with direct links to their purchase pages on Amazon.com. You can peruse this list of books at [http://www.stcwdc.org/refbooks-1\\_answer.shtml](http://www.stcwdc.org/refbooks-1_answer.shtml).

Books purchased through these links or by making a selection in the Amazon.com box on our site provide us with up to 15% of each sale,

which is used to help support the scholarship fund for the Austin T. Brown Technical Communication Scholarship Competition for high school students and the Shirley G. Carter Memorial Scholarship for college undergraduate and graduate students. Proceeds may also support chapter operations such as printing and mailing.

To help the chapter, all you must do is enter Amazon.com through the link on our home page or elsewhere on our site, and add items to your shopping cart within 24 hours of your arrival at Amazon.com via the link. Even if you don't purchase the items in your shopping cart immediately, as long as they are added to the shopping cart within that first 24-hour window, they contribute to the chapter's referral bonus if you place your order before the shopping cart expires (usually after 90 days).

Thanks in advance for your support of this program!

*[Continued from page 1](#)*

- Translate a business case into an IT plan
- Develop logical, quantitative user processes and roles
- Apply the HPD to interface design and product rhetoric, including instructional and persuasive text

### The Risks of Goal Re-interpretation

Usability testers, proposal writers, program managers, technical writers, and customer support providers develop their own interpretations of how to help users get what they want by interacting with a product. Unfortunately, as business ideas become projects which become prototypes which become a product and its user guidance, the interpretations necessary for this progress for each of these stages are re-interpreted again and again, causing unnecessary layers of abstraction and assumptions.

The assumptions underlying these re-interpretations are rarely recorded, causing other developers and supporters to promote false and misleading assumptions about the purpose and intent of the product, creating additional re-interpretation and a degraded relationship between the product and the goals that support it. HPD can help declare these assumptions and minimize re-interpretation.

### The Consequences of Not Reporting Design Assumptions

Chains of re-interpretations threaten the original intent of the design. Without the accountability of recording design assumptions, the responsibility for matching the product to business goals can remain vague and undefined, which can be an advantage for an individual stakeholder but can yield definite negative outcomes for a product.

While it is possible to contribute to the development of a product without assuming the greater purpose of the product, on the whole developers make assumptions about the purpose of product components throughout the lifecycle. Not holding developers to these assumptions frees them from responsibility for their assumptions when they make products that do not fit their intended purpose.

Products without declared interpretations of business goals can pass quantitative inspections but fail to achieve marketplace or client goals. Stakeholders who develop products without declared interpretations of the product's business goals can risk time, effort, and job security on products that have a negative effect on individual and organizational income, security, and prestige.

### How Hierarchical Process Design Protects Original Design Intent

HPD is a method that directs the design, development, and communication about a product from conception to end-user interaction. HPD declares the assumptions about the purpose of the product from the beginning and provides a framework for how to develop the product and how to communicate about it. HPD enforces a higher standard of accountability by requiring stakeholders to be accountable for the assumptions between the business goals and the design plan.

### Hierarchical Process Design in Four Easy Steps

1. Build a Goal Hierarchy for your product
2. Build flowcharts for every process (every node at the bottom of your Goal Hierarchy)

3. Build a table that lists every product state for every node in every process flowchart
4. Build, cost, explain, test, and improve your product based on the Goal Hierarchy, the process flowcharts and the table

These steps are explained below.

### The First Step: The Goal Hierarchy

To explore this model, write the goals for an IT product that you manage, develop, maintain, or explain in text or some hypermedia, like online help or web content.

If you have an IT product for which you would like to build a usability testing plan, hopefully you

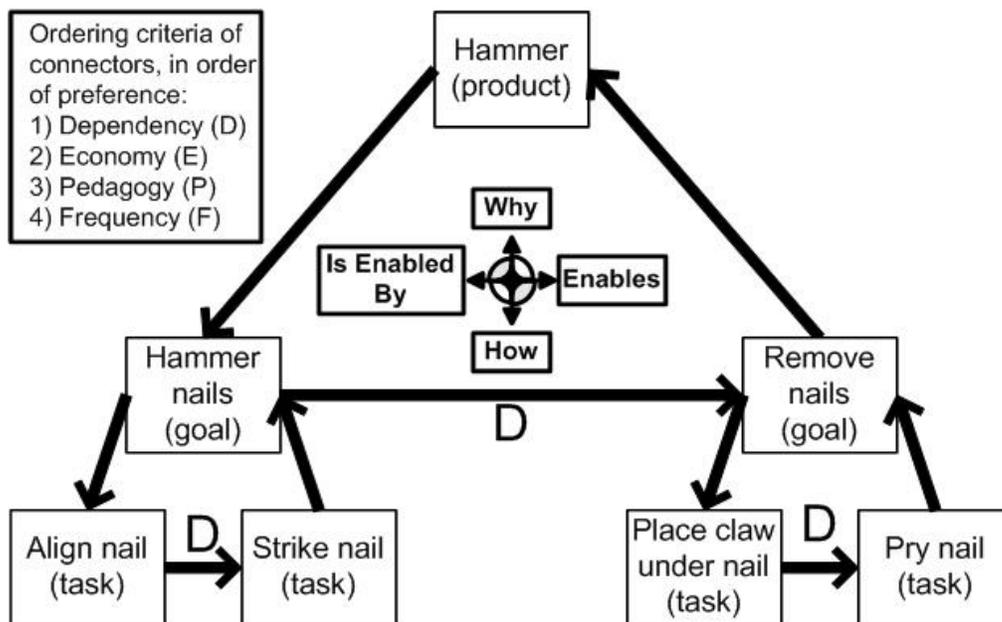
have notes about this product for your own reference so you can build a usability plan from the models in this paper.

### Where Goal Hierarchies Come From

Writing can use the same syntax to record the history of how a third party solved a problem and explain to a reader how to solve a problem. Unfortunately, few models in psychology work the same way whether the researcher is recording how a subject in an experiment solved a problem or creating instructions to solve a problem (like creating artificial intelligence). One model that works the same in either case is the goal hierarchy.

Because goal hierarchies use the same syntax in both recording and directing problem solving,

## Goal Hierarchy for a Hammer



These bottom nodes are also Processes because they have no subordinates. They have no subordinates because they can be flowcharted with binary logic

they provide a lot of power for software design, because software both creates a problem-solving environment and needs to be tested for its ability to help people solve problems. Goal hierarchies combine the explicit linear sequence of text with the detail, accuracy, and precision of nonlinear representations. With a goal hierarchy, you can build and test from the same tool.

A goal hierarchy defines a product as a set of goals in order of how they relate to each other. One goal hierarchy can guide and explain many dimensions of the same product including:

- The purpose of the product
- How the components of the product relate to each other
- How people use the product
- How to design the product to serve the user goals
- What user guidance to write and how to order that guidance, including training, online help, documentation, and customer support

Using a goal hierarchy can help designers monitor all of these dimensions using one model, and direct changes throughout all of these dimensions from a single source. This single-source method of defining the purposes of a product helps to build a product with provable functionality, minimal technical complexity, and a framework for building consistent messages through all the channels people use to communicate about the product.

Although goal hierarchies represented in text are visually similar to a classic writer's outline, goal hierarchies have tighter rules that define what goes where and in what order, and the ordering criteria can be difficult to represent in text. For example, here is a text representation of the goal hierarchy above, with its ordering criteria removed:

Again, the above text representation looks like a text outline, but reflects stringent hierarchical

### Subject: Hammer

Goal 1: Hammer nails into wood.

Task 1.1: Align nail with surface.

Action 1.1.1: Obtain nail.

Action 1.1.2: Place nail perpendicular to surface.

Task 1.2: Strike nail.

Action 1.2.1: Hold nail upright, keeping fingers away from nail head.

Action 1.2.2: Strike nail head with hammer head, repeating as necessary.

Goal 2: Remove nails from wood.

Task 2.1: Place claw under nail.

Action 2.1.1: Align edge of hammer claw with nail head.

Action 2.1.2: Slide claw under nail head.

Task 2.2: Pry nail.

Action 2.2.1: Brace hammer head against surface.

Action 2.2.2: Pull hammer handle down toward the surface, in the direction of the hammer head.

ordering criteria of order (what goes first and why), ordination (what is above and below what), criterial equivalence (what subordinates are necessary to yield the equivalent state of their superordinate) and inclusive and exclusive (meaning each element defines its meaning for the entire goal hierarchy and does not share meaning with any other element except in a superordinate relationship above its subordinates).

### A Problem of Order

The most subjective task in building a goal hierarchy is determining the most useful order of goals, tasks, and actions. Different criteria can affect the order, and they have the following order of preference:

1. Dependency (users must perform A before B)
2. Economy (performing A before B saves money, time or effort)
3. Pedagogy (learning A maps onto learning B more than the other way around)
4. Frequency (users perform A more often than B)

Sometimes, stakeholders will plead for an exception to this ordering criteria, but the desire for this exception is usually based on an unspoken perception of potential personal or organizational loss, which suggests a sorting criterion the stakeholder has yet to define for themselves or others. These requests for exceptions reveal unspoken assumptions. Record these assumptions as evidence to support the goal hierarchy you build.

The above criteria are in order of preference because an earlier criterion will often include the following criteria as well, e.g., ordering by economy often yields a valid pedagogical ordering and frequency ordering as well. If you are designing an existing product, you can interview the end-users of the product for the

goal hierarchy baseline. If you are designing a new product, you can interview the end users who are using current competing products or the product your new product is intended to replace to build the goal hierarchy baseline.

### Detail versus Abstraction

Goal hierarchies that are too detailed can be too difficult for the user to match the goal hierarchy to the task at hand. Eventually, all goal hierarchies, regardless of their original purpose, can be subordinated into discussions of theoretical physics or superordinated into discussions of metaphysics.

An analyst can continue to ask and answer "How?" questions about the lowest elements in a goal hierarchy until the bottom level of the hierarchy is explaining the atomic behavior of the atoms in the organic compounds in the cells of the user's brain that trigger an electrochemical impulse to the muscle nerves in the finger that presses the Enter key that confirms a command in a user interface. This level of detail is rarely useful.

Conversely, an analyst can continue to answer "Why?" questions about the top node of the hierarchy until the hierarchy is expressed as a product of the will of (metaphysical entity) for the user to be happy, safe, or fulfilled in some way, which is the reason for working at the job that requires the use of the software product in question. This level of abstraction is rarely useful.

### When the Actions Become a Script, Don't Push Any Further

Although a lot of writing is based on a hierarchy of information, such as classification systems in science and law, people still experience information one word, image or sound at a time. Writers and editors struggle to make complex sets of ideas fit into a linear order so that the text shares the nonlinear relationships of the concepts in the text.

Goal hierarchies help overcome this struggle by taking the overall subject, breaking it down into

discrete elements, taking the smallest elements at the bottom and explaining them as flowcharts of behavior, with the complex relationships to other elements explained at each state of the path.

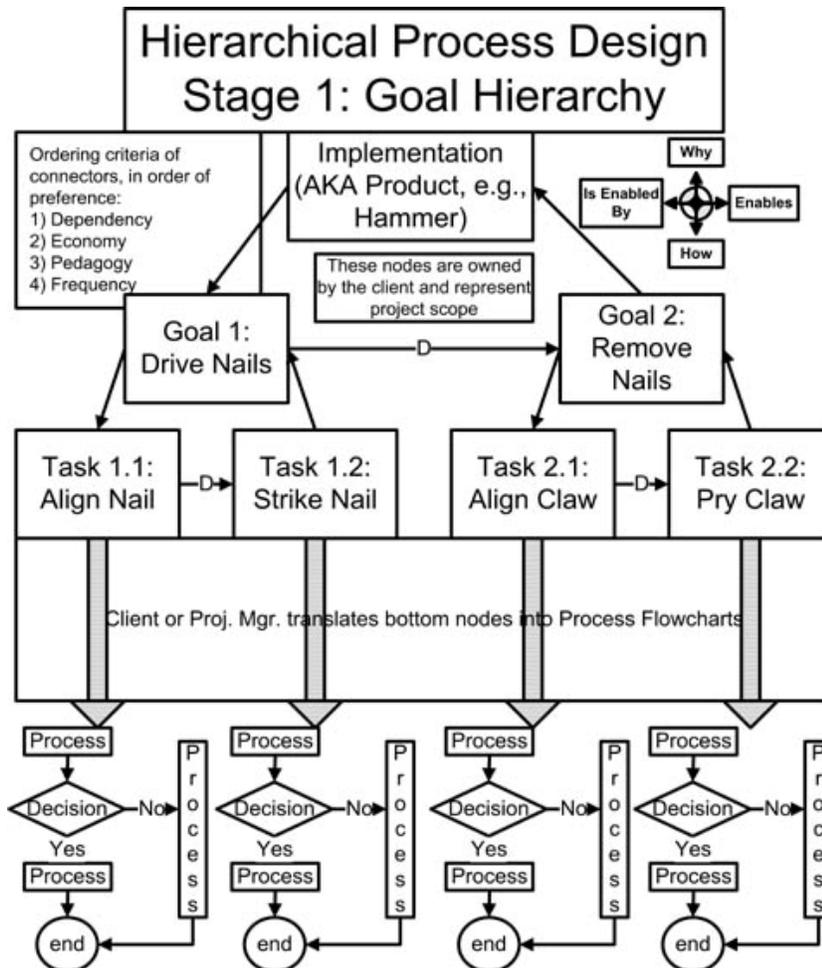
For example, to buy groceries, you have to have money, go to the store, select the groceries you want, and pay for them. To explain buying groceries, you could list these activities as goals (obtain money, go to store), break some of them down into subgoals (walk or drive?), and flowchart each of the subgoals.

That way, when you write the story about selecting the groceries you want, you can get to the part of the story when you're comparing each purchase to your budget, which relates to the goal of obtaining money. Because you're

breaking down the main project (buying groceries) into the smallest elements you need to explain those elements as stories, you're creating a method of explaining your project that's as linear as possible without limiting what you can say or how you can say it.

### How Goal Hierarchies Represent Problem Solving

When the goal hierarchy is subordinated sufficiently so that all of the nodes at the bottom can be defined as flowcharts with binary logic, these nodes fit the criteria of processes and can be defined as process flowcharts. The graphic below depicts this process.



### The Second Step: The Process Flowchart

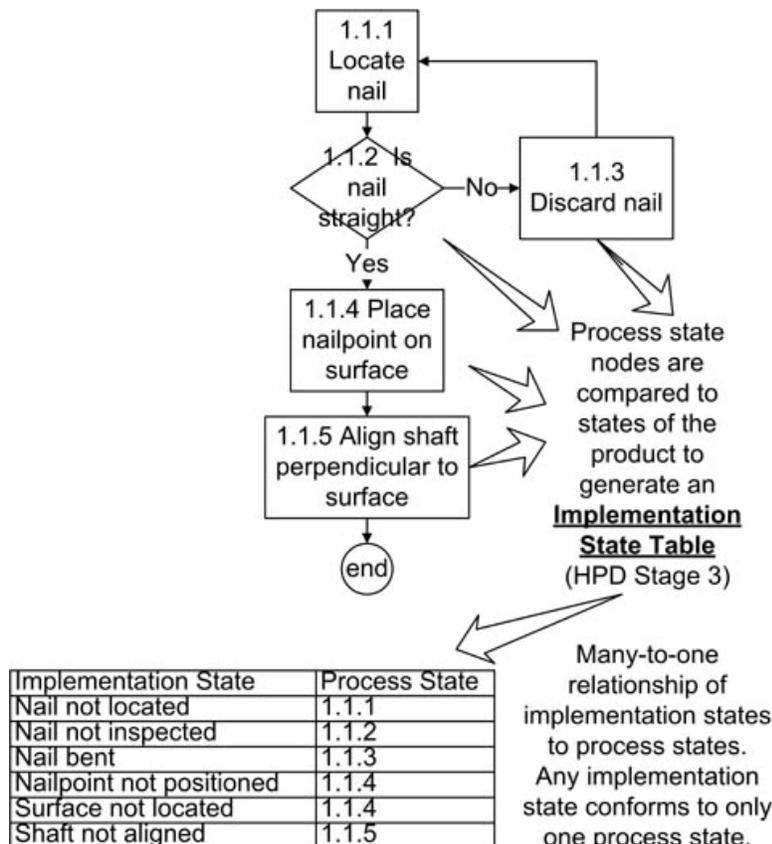
The process flowcharts define the processes of the goal hierarchy into flowcharts with binary logic so the product can be designed, tested, and improved to match the states of the processes from the perspective of users. Each state in the flowchart also declares the role responsible for that state if there are multiple roles in a flowchart.

### Risks and Returns of Process Ownership

Discretionary authority of the content of the process flowcharts can be owned by the project owner or the project manager, and there are competing priorities that guide which role should have this authority.

Flowcharts owned by the project owner (the owner of the overall business change that is being supported by the product) allow for better process control, risk management and process integration, but can cost more money to implement because the project manager does not have direct authority to change the process to meet the business outcomes of the product. Flowcharts owned by the project manager (the person who directs resources so the product enables the business change) allow for better implementation efficiency because the project manager can adjust the business processes, but can generate more integration expense due to integrating the product into a different business process. There is also an increased risk that the product will not meet the business outcomes of the project owner.

Hierarchical Process Design Stage 2: Process Flowchart (Task 1.1: Align Nail)



### Using Process Flowcharts to Drive Usability Test Designs

The separation of a product into manageable goals, tasks, and processes provides the usability test designer with a framework for assigning subsets of the product to individual tests, what to test in what order, and an explicit representation of the state of the product and the state of the user processes at every state of the test.

As designers build goal hierarchies to design product interfaces and rhetoric, they can encounter a set of actions that have to occur in a precise order. For example, if a designer is explaining how to add a date field onto a document, the set of actions could be:

1. Select Insert from the menu bar.
2. Select Date and Time from the Insert menu.
3. Select a date format.
4. Press the OK button.

This process is much easier to document than trying to explain all the ways you can use the Date and Time function. The goal hierarchy makes it easier to determine what paths have to be documented. It also makes for instructions that are easy to improve on later because all you have to do is figure out how big a branch of the goal hierarchy you want to test, tell people to follow the instructions for each path in the branch in the order that's already defined, and record how long they took and where they failed to complete a process.

This method also improves customer service, because customer service and customers share a common method of getting what they want by following a process based on user goals instead of the peculiarities of the tool. Customer service research can record where users were when they failed to proceed (called a "point of failure"), and associate these points of failure with call frequency, support expenses, and the returned value for improving the product where the points of failure occurred.

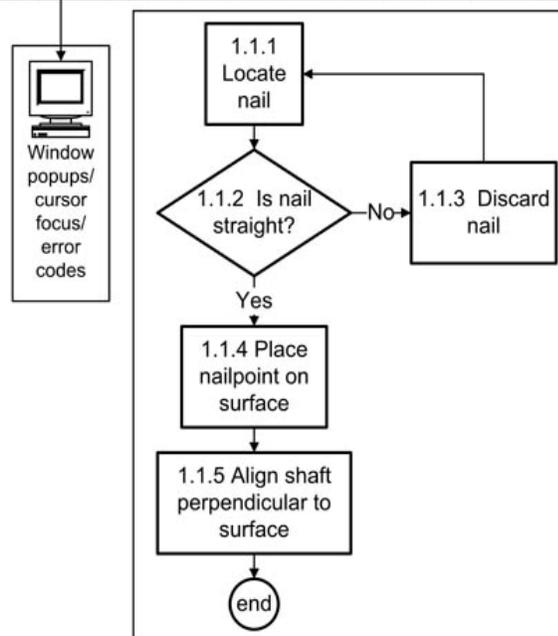
### The Third Step: The Implementation/Process State Table

Implementation states are inventoried in this model, and costs and resources are associated with process states and their corresponding implementation states. This allows designers and managers to cost and budget against the real value of advancing the business process instead of the interpreted value of advancing the state of the product.

Process states are owned by the project owner. Implementation states are owned by the project manager. Each process flowchart reveals the next human action that must occur to progress in the current iteration of the process, and the role of that person if roles are defined in the process flowchart.

Hierarchical Process Design Stage 3:  
Implementation/Process State Table (Task 1.1:  
Align Nail)

Implementation State	Process State
Nail not located	1.1.1
Nail not inspected	1.1.2
Nail bent	1.1.3
Nailpoint not positioned	1.1.4
Surface not located	1.1.4
Shaft not aligned	1.1.5
* only one state per implementation *	one or more implementations per state



### Using Implementation/Process State Table to Drive Product Design

Product users and stakeholders can use this table to identify specifically what business action has to happen next, the business role responsible for that action if roles are identified in the process flowchart, and the implementation states that support each business action. Every implementation state is matched directly to the specific state of the business process it supports. Because every process state has an explicit relationship to its process, task, and goal, every implementation state has an explicit quantitative relationship to the business goals that drive the entire product. At this phase product testing, usability testing, product documentation, and product support heuristics can be created or changed so that all communication regarding the product has a unified structure based on a single plan. The state of the product exclusively

identifies the state of the business process.

### Hierarchical Process Design Deliverables

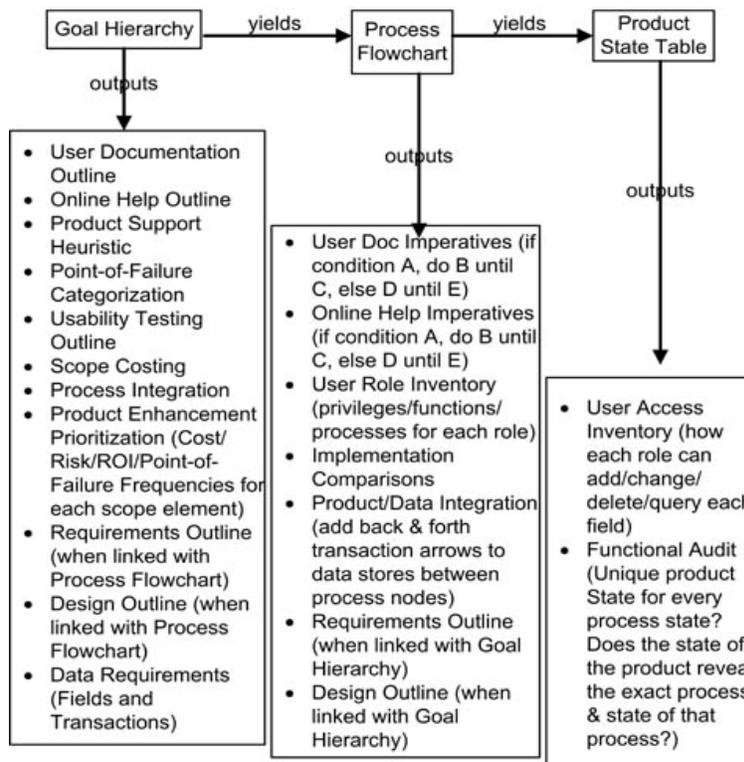
*(Please see diagram below)*

### The Fourth Step: Integrating HPD Into the Product Lifecycle

### Create the HPD Baseline

There is always a baseline for a business process. The first word processor competed in the marketplace with the typewriter and mimeograph machine. New IT products usually have an existing competing product or products that can achieve the goal hierarchy baseline of the new product. Changes occur in the process flowcharts, and improved outcomes and efficiencies in the process flowcharts are an effective way to demonstrate the advantages of the new design. Designers can also compare costs of each business goal in the baseline versus the new design.

Hierarchical Process Design Deliverables



## Obtaining Goal Hierarchies Through Interviews

Once you determine the product(s) that yield(s) the goal hierarchy for the new product, you can interview the users of those existing products to build the process flowcharts for the processes (bottom nodes) of the goal hierarchy. You can also ask questions that will yield process improvements in the new product, e.g., "What does the product you're using now make you do that the product should be able to do by itself?"

## Write User Documentation, HTML, Online Help, PDF

Because the goal hierarchy is a taxonomy with a hierarchy and order based on user outcomes, the goal hierarchy becomes the structure of the information about the product. Goals 1, 2, and 3 become Chapters 1, 2, and 3. Task 1.1 becomes Section 1 of Chapter 1, etc. The process flowcharts become the linear sequence of steps to accomplish the overall objective of each process, and each flowchart node can be described by changes in the product state with imperative sentences, descriptions of the change in the product state, and screen captures before and after the change in state.

## Design a Customer-Support Heuristic

The customer support heuristic follows the structure of the goal hierarchy by asking questions that locate the process the user is trying to achieve and the state within that process in which they failed to advance. If you design all product rhetoric on your HPD, support personnel can direct users to the location in all forms of the product rhetoric that address the process state the user is trying to achieve. Support and enhancement costs can be tracked by each process state, process and goal, and support personnel can track point-of-failure expenses for each process state.

## Create Usability Tests

Testers can take a manageable subset of the goal hierarchy and test the ability of users to advance through the process states for each process, and generate frequency and time delays for points of failure for every process and process state.

## Product Design

A goal hierarchy can direct the development of prototypes before the physical features and user interface of the product are determined. Defining a product by its goals and tasks allows the usability of the product to be evaluated and changed from the beginning of the lifecycle, as opposed to defining products by functional specifications, which tends to forestall usability testing until late in the lifecycle.

The goal hierarchy can also expose preconceptions and assumptions that the designers have that might not otherwise be explicit. If a design team was developing a hammer and had generated the goal hierarchy above, an analysis of the hierarchy reveals that the team has assumed that the hammer they are building is a claw hammer; otherwise, they would not have been able to solve for the goal "Remove nails" using the subordinate task "Place claw under nail."

Organizations that are building products for a client can use a goal hierarchy to illustrate the progress of development, the allocation of resources to each goal or task that the product will help users accomplish, and how the components built or planned (e.g., the files and transactions of a software product) relate to the user goals of the product.

Because designing software has unique challenges and tools compared to other product development, the software industry has developed several ways of measuring and predicting the relationships of expenses, resources, and the components of a software product. One way of measuring these relationships is to use formulas that assign

numerical values to the logical records of data and transactions required to fulfill a set of user requirements, and derive an overall mathematical value of the functionality of a project. A dominant model in the industry for generating these values is function point analysis. The product example described in this paper has been subject to function point analysis, and was described in its accompanying user documentation using goal hierarchies. A possible way of integrating the metric effort and the rhetorical effort would be to sum the function points associated with each task and goal on the goal hierarchy. Associating the function points (and therefore, the time and expense) for each task and goal would provide clients with specific information about the expense and resources for every enhancement as the client adds new requirements to the functional specifications.

### Discussion

Determining the hierarchy and order of product interaction is important to product design and instruction design for a similar reason: there are dependencies and relationships among interactions that encourage one particular order over another for reasons of dependency, economy, pedagogy or frequency. Many work tasks conform to a process that can only be accomplished in one specific order, because the beginning state that the object to be operated on to perform task X only occurs as a result of completing task X-1.

Some product interactions do not have a natural, or even an obviously preferable, order among them. Some task sets occur in a sequence that is typical only because of tradition. Some tasks are ordered in terms of their predicted or explicit importance to the user, much as a newspaper reporter might order paragraphs by supposed

reader interest so that the editor can delete paragraphs from the bottom for length. An instructor or writer can also order the instruction of user interactions by the transferability of the learning required to accomplish each task to the rest of the tasks in the set. In this way, the document design not only incorporates an understanding of the state of the user and the product at the outset of each task, but also incorporates the change in the knowledge state of the user as they learn each task.

### References

- Bull, Kay S. & Kimball, Sarah L. *Online Teaching: An Instructional Hypertext*. Oklahoma State University, 1999.
- Fraser, Martin D., Kumar, Kuldeep, & Vaishnavi, Vijay K. *Strategies for Incorporating Formal Specifications in Software Development*. *Communications of the ACM*, 37 (10), pp74-86, Oct 1994.
- Jackson, M., *Towards coherent pluralism in management science*, *Journal of the Operational Research Society*, 50(1), 12-22, 1999.
- Wilson, B. & Cole, P. *A critical review of elaboration theory*. *Educational Technology Research and Development*, 40 (3), 63-79, 1992.

### About the Author

Basil White is an Information Technology Specialist with the Cyber and Information Security, Department of Veterans Affairs, Washington DC. A previous version of this paper was published in the 1996 Proceedings of the Annual Conference of the Society for Technical Communication.