

Interface Design as Cognitive Engineering

The design of a financial system interface is factor that enables but also limits the productivity of everyone who uses it. In didi's quarter century of experience in financial technology, we've noticed that the majority of financial systems are designed without adequately addressing the most important aspects of business workflow: the subtle steps in the thinking process an expert develops throughout his or her career. As a result virtually every revenue-producing role on Wall Street is hobbled by the systems meant to support it.

Common practices in interface design are far from best practices

Interface Design is all but ignored in financial systems. It's commonly expected that a functional specification will capture all necessary workflow issues, and that design is at best a cosmetic polishing that might be useful for client-facing applications but adds no value to internal systems. There is nothing farther from the truth. It's our experience that virtually every work process on Wall St. can be made significantly more effective.

Properly practiced, design is not just fashion or branding but an optimization process whose objective is to connect the information in the computer with business tasks in a deep and effortless way. Seen as an optimization process, it's clear that we can focus on the working parts of the information transfer between computer and financial expert. But the skills needed to understand those working parts are not taught in business schools or computer science curricula.¹ In a typical development cycle, a business analyst is asked to interview traders (for instance) to develop a list of use cases, a dictionary of the data needed to support them, then write this information up for a development team to code. This is an important step in defining *what* the system must do—but it does not capture *how* the traders work at a fine enough level of granularity to facilitate the real business processes: what really goes on in their minds.

Design as optimization

To optimize a system's interface we need to dig deeply inside the thought processes that are the essence of each business task. We need to support every sub-second cognitive step with data (or input opportunities) in exactly the right form to fit how it is being thought about at that moment. It also needs to fit all the other ideas that are part of that task. Traders themselves do not have access to how those processes work any more than a grown adult can describe in words how to tie a shoe or ride a bike. The processes have become "automatized," so they don't come out in interviews.

Engineering is applied science, and presumes an almost cookbook-like methodology to apply to problems, at least as a starting point. At *didi* we directly apply science and have developed such a methodology. The results are as remarkable as one might expect where no engineering has been applied before. Applying engineering practices from the wrong domain (e.g. Computer Science) can sometimes be worse than no engineering at all: human minds *do not* operate like computers.

The sciences we apply range from Visual Perception and Cognitive Science through Psycholinguistics.² The cookbook-like methodology creates a working process that reveals to all participants exactly what progress is being made, and how obvious its absence was in retrospect.

¹ Except Columbia University's Department of Computer Science, where *didi* principal W. Bradford Paley was asked to teach the *didi* design methodology in graduate seminars.

² See the "paper lecture" *Interface and Mind*, available from *didi*, for more details about how these sciences from the basis of cognitive engineering.

Introduction to a rigorous design methodology

An introduction to this methodology takes a 14-week college class (or a three-day intensive seminar at a financial firm), but some of its key underpinnings and findings can be hinted at here.

First, a thorough understanding of every element in a business role is compiled, including:

- 1) Management/firm goals, working expert's goals (e.g., a trader's goals), the tasks that build toward the goals, and sub-second-scale cognitive steps necessary to complete the tasks
- 2) The entities involved in a task (e.g. order, symbol, market, algorithm)
- 3) The relationships among those entities, and the transitions they go through
- 4) The business meaning, value, risk, and urgency of every task and transition

This exhaustive dictionary of things/processes/issues is then partitioned—not following database idiosyncrasies or within-firm silos of practice—but following business practices and needs. Thus a single screen might need data feeds from four or five different databases or even different business groups to accomplish a specific task. Don't send the expert to five screens: bring the data to one.

There are techniques to turn this partitioned dictionary into candidate designs, e.g.:

- 1) Tasks become screens: each one- to five-second task initially gets its own window in which we can create a tool perfectly honed for that task. (Imagine using a Swiss Army Knife to build a home—financial systems typically throw a database into a do-everything table and let the user configure what they want for various tasks.)
- 2) Entities become objects in those screens. And importantly, the objects are designed to have the exact and minimum amount of information in them tailored to the task at hand.
- 3) Relationships become annotations or interface behaviors. States are indicated by coloring, shaping, expanding, or re-positioning of objects. (Color-coding alone is not nearly enough.)
- 4) The business meaning drives the layout of the windows, while the value risk and urgency drive the visual layering within each window. Information is where it's expected; more visible the more important it is.
- 5) Windows and tasks are then re-integrated into the whole workflow of the expert, following both business and local goals. Sometimes this allows task-centric windows to be combined. Even where screens stay separate and optimized for a given task they can be tiled into a worksheet or pop-up only when necessary. And they are *always* tied together with visual techniques that let the eye easily find the same and related objects in nearby screens.³
- 6) Later steps in the methodology do step-by-step testing of the system well before coding starts by using "paper prototypes"—with huge cost and time savings.

How many financial screens exhibit the simple mapping that comes out of step 4: that the most important business issue is the most visually obvious thing on the screen? How many screens are explicitly tailored to ease the mind's task-switching costs as an expert goes from one to another, as addressed in step 5? Standard UI guidelines address none of these issues.

Literally hundreds of techniques exist to tie on-screen visuals to mental processes. Perhaps the strongest value comes from the fact that screens designed to reflect the decades-honed best processes of the experts actually guide every new user to think like your most productive people. The benefits are concrete and create significant value in many dimensions; including productivity gains, lower training costs, risk reduction, and lower support/maintenance costs. Most important it results in improved morale and creativity in your best people because the system finally lets them think not as data entry experts but as the business experts they signed up to become.

³ See the keynote talk-related notes *MoMA Desktop/Data Rain: InfoVis as Design in an Art Museum*, also available from *didi*.

Example Interaction Design Valuation Worksheet (ask us for a working one)

This spreadsheet gives an example of the total value generated for a single project (here, redesign of a trading desk at a single bank location). Note the over \$16 million result.

Improvement scenarios				Final yearly value of each scenario		
Minimum	Expected	Optimistic	Minimum	Expected	Optimistic	
Productivity			Improving workflow efficiency			
Task			Fully loaded cost/person			
1.0%	2%	5%	Sales trader	\$1,700,000	100	
1.0%	2%	5%	Position trader	1,700,000	50	
Risk Reduction			Reducing risk for revenue streams			
Kind of risk			Cost of each error			
20%	40%	80%	Break	5,000	100	
20%	40%	80%	Break: Image risk	50	100	
5%	10%	20%	Missed deal (e.g. too late)	1,000	500	
20%	40%	80%	Delayed or killed project	3,000,000	0.25	
New Business			Enabling business opportunities			
Opportunity type			Value per opportunity			
0.0%	25%	100%	New customer acquired	500,000	5	
0.0%	25%	100%	New trading strategy	2,000,000	0.5	
Visionary Stuff			Higher-level repercussions of visually supported analysis			
Opportunity type			Value per opportunity			
90%	50%	100%	Time spent training	141,667	15	
0.0%	25%	100%	High-level creative ideas	500,000	4	
Newly Created Value			Time horizon (in years) 4			
			Value created first year			
			Value over 4 years			
			Discount to present value (simple linear discount)			
			\$3,038,500	\$8,089,500	\$21,266,500	
			12,154,000	32,358,000	85,066,000	
			(6,077,000)	(16,179,000)	(42,533,000)	
			\$6,077,000	\$16,179,000	\$42,533,000	

Typical Problems Solved by *didi's* Interface Design Methodology



This list of common concerns may sound familiar. They are *not* a natural part of the financial technology development process. They are symptoms of an inappropriately low investment in the training and time needed for proper interface design. Cognitive Engineering leverages the knowledge and practices of a distinct and well-developed discipline that is rarely represented in typical financial user/analyst/developer system specification teams. Plugging that gap directly addresses and solves many problems.

Business-side / IT developer relations are typically improved, allaying these issues

- 1) After a financial system is coded it needs major revisions before it can be used by business people (an embarrassingly large fraction of projects are just terminated)
- 2) Business people are frustrated because the technology group doesn't seem to understand the business needs
- 3) Developers are frustrated that they are constantly re-writing code based on changing specifications

Mapping cognitive business workflow to system design eases these frustrations

- 4) Business people find it hard to switch from another firm's system
- 5) Business people blame the system for errors or inefficiency
- 6) Business people feel frustrated that a seemingly simple operation is difficult to execute in the system
- 7) It is not possible to accomplish a simple business goal without several systems
- 8) It is hard to find where in the system one needs to go to accomplish something

Sophisticated information layering simplifies & clarifies, addressing these problems

- 9) Important business events (alerts, errors, opportunities) are missed even though the information is in the system
- 10) Screens look too busy, or complicated; there is "information overload"
- 11) It is hard to find certain information on a screen even though it's there
- 12) One needs to go to another screen or system for information in the middle of a task in order to complete it
- 13) Unnecessary information is displayed

System complexity can be tamed by combining design and prototype walk-throughs

- 14) Clients need help understanding how to use a screen
- 15) Clients take business elsewhere because screens are hard to use
- 16) Clients ask for "additions" that are already a part of the system
- 17) Business people need paper "cheat sheets" to accomplish tasks

At *didi* we're happy to meet with you to address these issues with respect to your own specific projects. We can detail exactly how our design methodology can reduce or eliminate them—it is not a mystery, or personality-based talent, but a straightforward, teachable process.

The Missing Link: *didi's* Role and Deliverables in Optimizing Business Workflow

didi

Few financial systems are as satisfying or easy to use as an iPhone or even Quicken. This is simply because there is an entire discipline missing from most technology development processes on Wall Street: a rigorous, thorough, fine-grained analysis of the cognitive steps business people go through as they accomplish their most important tasks. This iterative analysis/testing process tightly links business workflow and functional specifications—making sure the business flows as easily as the data does.

We augment your existing team with the missing skills.¹ Then we deliver results in a directly usable form: completely worked out screen designs, ready to be included in the specifications that are handed to developers. During your implementation² we help redesign where necessary and mentor sophisticated graphics techniques if needed. We also help with fine-tuning after initial deployment.

We can do this because a quarter century on Wall Street has taught us how to integrate several fundamental skill sets that are rarely unified in any one team or person—all of which are critical to proper interface design:

- 1) *Financial fluency*: an understanding of the business
- 2) *Cognitive science*: an understanding of how the mind integrates information to make decisions and act on them
- 3) *Typography*: an understanding of how to structure screens so they are readable, recognizable, comfortable, and obvious
- 4) *Illustration and graphic arts*: the ability to make different kinds of data look different and instantly be recognized; respecting branding guidelines
- 5) *Information visualization*: the ability to design graphical representations of data to condense, contextualize, and highlight important information
- 6) *Economics*: the ability to explicitly and rationally do a cost/benefit analysis that drives prioritization of which features to develop and when; or even when *not* to spend time and money on design
- 7) *Computer Science*: the ability to recognize programming-, communications- or compute-intensive features, to avoid or help optimize them
- 8) *Software engineering*: the ability to train developers how to implement sophisticated custom graphics, behaviors, or widgets
- 9) *Testing*: understanding of how to create & use fully-engineered “paper prototypes” to completely test a system before coding starts

¹ Given enough organizational commitment (typically a full-time mid-level to senior person with both business and technical experience) we can even teach the process to your team, completing the technology transfer.

² We can also source best-of-breed graphics and code production with our strategic partners. Often it makes sense for a firm's internal team to remain focused on business logic and outsource look-and-feel, skin, widget or graphics development.

These skill sets enable several complimentary processes in *didi's* own rigorous and carefully worked out design methodology.³ Each process addresses the interface from a different perspective; all are necessary for an optimal system:

- 1) Business workflow analysis
- 2) Business process re-engineering
- 3) Audit of all tasks related to a specific business role
- 4) Low-level cognitive deconstruction of steps needed to accomplish each task
- 5) Identification of all input and actions needed for each mental step
- 6) Layout of screens optimal for each task
- 7) Illustration of every business concept so that they are easy to distinguish
- 8) Iterative testing and tuning of the design with real business experts using paper prototypes
- 9) Support of a firm's creation of development specifications
- 10) Mapping of an "ideal interface" design back to programming and budgetary realities by changing scope and level of design detail, or identifying phases for staged development and deployment

Many of these processes are partially or fully addressed by existing project development teams, but two perspectives are rarely included: numbers 4 through 7, and number 8.

Step number 8, testing with *paper prototypes*, is a technique that has recently been developed and is taught incompletely in very few places. But it can save calendar months and large fractions of development budgets because it catches problems before they make it into code—allowing hundred-fold⁴ reductions in the effort and cost spent on missed or misinterpreted system functions.

And the steps relating to cognitive process mapping and screen-based illustration of the results, 4-7, are rarely included because they require a significantly different background than most development teams possess. Rigorous design is a distinct discipline. You can put your best business analyst, trader, graphic artist, and developer in a room and still come up short because those disciplines never prepare people to dig for and codify task-related thought processes. But it's those thought processes that keep the firm afloat.

Addressing this missing link captures significant value. We at *didi* are happy to walk through specific projects with you to detail how we can reduce costs and increase productivity by becoming part of the team.

³ Columbia University's Department of Computer Science found this methodology valuable enough to install *didi* principal W. Bradford Paley as an Associate Adjunct Professor expressly to teach it in graduate-level seminars.

⁴ The (real!) hundred-fold savings described: it's approximately ten times faster to fix a problem in a paper prototype than in a specification, and ten times less effort to fix a specification than to fix the system after it's been coded.

Selected Recent Clients, Projects & Testimonials

Goldman Sachs: GS NYSE Broker Handheld

Key feature: Invention of “information object” representation to make interaction with orders effortless.

Richard Genna, then head broker for GS: “We’re getting more information in real time as opposed to less information with a lag time.”

Andrew Silverman, then broker for GS: “It’s hard to let go of something you’ve been doing for a long time and do something new. Brad’s given us an ergonomic shorthand. You don’t need to learn it, you just grab it with your eyes. The bottom line is that it makes the broker more productive. And that makes the firm more productive.” (Both quotes from I.D. Magazine’s June 2000 feature article on the project)

Morgan Stanley: Foreign Exchange system (both sales and trading)

Key advantage: Created “unmissable” alerting visuals within the trading blotters: dangerous situations were made impossible to miss by visual layering; the risk of trading with a client at a deficit was reduced by introducing a real-time market feed that modified the trading buttons

Merrill Lynch: Equities Sales Trading System

Key advantage: Created clear visual trade tracking within the sales trading blotter: every trade’s amount filled and state was immediately visible; “unmissable” alerting visuals

Senior Project Manager: “I worked with Brad at Goldman Sachs and thought his jet-cockpit-like ergonomic approach was normal—until I tried to find a team in London to do the same. We looked at everyone we could and didn’t find a fraction of the background, insight, or capabilities. I was happy to be able to use *didi* once I got back to the States. The resulting interface clearly made our Equities Sales Trading more productive as well as happier using the new system.”

Morgan Stanley: Internal organization chart, HR, access control system

Key transformation: Extremely abstract representations of complex relationships among teams, people, roles, securities, systems, clients and other entities were made obvious and drag-and-drop easy; 50+ Web-based screens were reduced to to six objects & three views

New York Stock Exchange: Specialist Workstation for Hybrid Market

Key advantage: Put up to four times as much information in the same screen space, more readably if information is measured by *characters* on the screen; perhaps over ten times as much information on the screen if measured as *inferences* that the specialist can make

Lou Pastina, Senior Vice President: “Brad Paley looks at the world differently, and provides a perspective in design that is innovative and intuitive for users. He is able to combine his unusual perspective with years of proven successful design and implementation to deliver transformational change with the mastery of a minimalist.”

New York Stock Exchange: Broker Handheld (design and prototype)

Key advantages: Sped up brokers a factor of fifteen or more (from 7 seconds to less than half a second) in the key process of “refilling” an order; greatly sped up cancel-and-replace operations for moving one or more orders to a different price point

Gordon Charlop, then President and CEO of WJ Dowd: “Apart from his great skill as a designer, Brad has a unique ability to develop a design by listening and understanding what clients want and need. At the NYSE, Brad was asked to design a state of the art portable order management system. His constituency was a broad group of high level brokers with disparate needs. By combining his communication skills with his design knowledge, Brad developed a remarkable product.”

Lehman Brothers: Prime Brokerage Web site, client Report Access system

Key advantages: Simplified client and internal definition of customizable reports and automated report delivery, organized immediate access to financial reports

Senior Project Manager: "Our Prime Brokerage Web site was complicated but the *didi* redesign made it both simpler and clearer. The *didi* approach made our highly-functional (but complicated) client-access Reporting system much easier by applying unique *didi* information-object technology, and the paper prototype-based design process we learned from *didi* saved us uncountable development hours and business-side frustration."

Rosenblatt Securities: Custom Advanced Charting Package

Key advantages: Integrated interactive Java Applet charts quietly into a Web page; allowed them to zoom out of the page for closer analysis; innovated point-query and curve comparison techniques to reduce misinterpretation risk (based on visual/cognitive principles)

Dick Rosenblatt, founder and CEO Rosenblatt Securities: "The Rosenblatt Securities business model requires an innovative, often unique, approach to the trading process utilizing the best technology. Brad not only takes the time to understand our ideas, but creates amazing technology to make them a reality. The result is functional, easy to use, and visually appealing. The fact that Brad is a pleasure to deal with, and works at close to light speed just adds to a positive experience at every level."

Earl Industries: Intelligence Community Analyst's Workstation Prototype

Key innovations: New interaction techniques allow instant association of disparate data objects in "blobs" or categorized relationships with links; the design allowed several analysts to work together, in the same room or globally; the concept of two-sorted multiple logical analysis was directly visually supported; scenarios organize information related to a developing "story"; and different stages of the formation of a story are visually represented in a kind of active audit trail

Ted Goranson, Project Director: "We came to Brad because common sense in cognitive flow is apparently rare. We needed someone to guide some conceptual thinking very early on in the project and have found his approach invaluable."

Rosenblatt Securities: Flexible report-generation code and TCA visualization

Key contributions: Compressed report length and made data more easily readable; developed clear typographic layout

BIDS Trading: Trade Blotter (trade input, tracking, alerting)

Key advantage: Combination of three screens/operations into one; visual clarification of the transaction-related business structure of a dark pool model to clients.

Other Testimonials

Legal constraints at some financial institutions prevented clients from going on paper record with quotes but they wanted to weigh in regardless. Here's what they said.

Head of Product Development, highly visible financial service provider: "Brad is unusual in this space in that he combines three often unrelated things: a very well developed design sense, a deep understanding of human cognition, and computer science. His work with us has had a remarkable transformative effect."

Project Director, consulting firm specializing in financial risk: "Words simply cannot do justice to what these techniques are capable of, and in a very practical vein how advanced this work is relative to the commercially available data mining / presentation software packages such as Cognos and Business Objects."

Senior Project Manager, Merrill Lynch: "Brad gave a seminar in the *didi* design methodology that left my development team inspired and motivated."