



## Pareto's Principle

### ***Applying the 80/20 rule for project scope and success control***

*Use scope definition to build relationships, not destroy them. Making considered decisions on how much to spend to get the right business benefit is a team approach when you try some old thinking in new ways.*

*Joe, CIO at ABC Tech, sank into a chair in CFO Ann's office. "I wish we had a more energetic team for this project. The implementation is so important, and we just got through defining requirements and people want to throw in the towel!"*

*Ann bristled. "Well, they'd be happier if your team wasn't already telling the users they can't have what they need. We haven't even gotten into prototyping and all we hear is that our requirements don't make sense."*

*Joe snapped back, "Well, if we had unlimited time, money and developers, you're right, we could build a miracle system that can make all the accountants obsolete!"*

*Ann stood. "This is the problem. Right here. How can you know what it will cost if you don't even listen?" She stalked out of her own office and slammed the door.*

### ***The Good of the Few...***

Even if their relationships aren't as bad as Ann and Joe's, there are few CIO's who don't know the frustration of a combative relationship with users during scope setting phases. Fear of overcommitting or letting users even think beyond the immediate to oversight expectations drives playing it cool and pushing back early. Business unit leaders get frustrated with IT, and reputations all around suffer. And it's all so avoidable, if we go back to basics.

In the early 1900's, the Italian economist Vilfredo Pareto identified that a "significant few" accounted for a disproportionate affect on the



## Pareto's Principle

### *Applying the 80/20 rule for project scope and success control*

---

whole. We know this observation as the Pareto Principle, or the "80/20 rule." Loosely translated, 80% of the result can be realized with 20% of the investment. And getting to the last 20% of results costs the additional 80%.

We all have seen the career limiting trend of large IT projects overrunning, scope creeping into massive cost overruns and the significant probability of being characterized as a failure no matter what goes live. And yet we know that if managed correctly, we can get 80% of business objectives from IT projects in the first 20% of the investment, and the rest can be managed to minimize just how much money we spend and risk we assume from the last 20% of benefit.

Using the 80/20 rule offers the single greatest opportunity to increase the probability of success in IT projects, and choosing when to apply it is even more important than the rule itself. IT can use the rule to use experience in a way that improves relationships and potential for success.

## *Applying to IT Development Projects*

### **Typical approach**

Let's go back to the typical approach to business requirements for an information systems development project. Systems analysts try to filter requirements right from the start, as a way to "manage expectations".

The analysts start the battle early, trying to minimize the cost of design and later development. They know that accommodating requirements they consider to be excessive (or downright outrageous) will only lead to having to explain to users just how much the budget will explode.

Users retaliate by defining all requirements as a "must have". Management reacts by freezing the numbers, users opt out, and no matter what eventually goes into production, no one is happy. So analysts try to head it all off at the pass, only succeeding in accelerating user disappointment.

The problem with the typical approach is that the user community isn't given the chance to make the business case for the requirements the systems analysts try to filter, sponsors aren't given the chance to



## Pareto's Principle

### Applying the 80/20 rule for project scope and success control

---

arbitrate, and IT isn't given the chance to show just how much CAN be done. The whole thing starts off badly and goes from there.

### Right Approach

So how does the "80/20" rule work here?

**First off, you do have to start with 100% to use it.** So gather requirements! All of them! No filtering or judgement at the start. Open the dialogue, gather requirements and let users make the business case for each one. IT's role here is to listen, understand, and capture. Providing a service in this way makes the whole thing positive for users, who get to explain what they want and ensure that IS is listening to them – for a change!

IT comes away with a complete understanding of the users' vision. From here, IT builds preliminary designs of the technical solution to deliver the features the vision includes. The design is the full court press of database designs, integration requirements, user interface designs, functional designs and technical specs.

Before you argue that this is a lot of investment for a solution that will never be built, recognize its immeasurable value.

**From this design, IT has defined EXACTLY where which requirements contribute complexity.** Causes and drivers of complexity give a fact-based cost to the benefit the users defined, feature by feature.

Armed with this information, IT can put together **two design scenarios**. Now before a full technical revolt takes place, remember where dollars go on IT projects. Saving money on design only spends it in development, testing, more development...you get the idea. So stay with us here.

The first design scenario is the complete feature set, with the full cost and timeline to develop. We can call this "**The Ideal Design**". If money and time were no object, this design would create the ideal user vision.

Then there is "**The 80/20 Design**". IT builds this design scenario by applying business knowledge and the understanding gained in the requirements gathering to get 80% of the vision into the design, finding less complex ways to get pretty darned close to ideal, but at a



## Pareto's Principle

### Applying the 80/20 rule for project scope and success control

lot less (maybe 20%) of the cost and time. To demonstrate this, the table below shows what that can look like in a management summary:

Major Project Scope Drivers	80/20 Design (\$K)	Ideal Design (100% Scope) (\$K)
Standard ERP Installation	\$3,000	\$3,000
Reporting: <b>Standard Reporting</b> <i>Custom Reporting</i>	\$100	\$500
Consolidation <b>Standard Financial Consolidation</b> <i>Third Party Consolidation</i>	incl	\$1,000
Chart of Accounts <b>Standard COA</b> <i>Custom COA</i>	incl	\$500
Product Data Management <b>Standard BOM Management</b> <i>Third Party Product Management</i>	incl	\$900
Inventory Routing <b>Standard Inventory Management</b> <i>Custom Inventory Tracking and Routing</i>	incl	\$1,000
Sales Commission Calculation <b>Interfaced commission data</b> <i>Customized commission transactions</i>	\$30	\$300
User Interface <b>Standard User Interface</b> <i>Custom User Interface</i>	incl	\$800
<b>Total Cost (\$K)</b>	<b>\$3,130</b>	<b>\$8,000</b>
<b>Duration</b>	<b>12 months</b>	<b>28 months</b>
<b>Risk</b>	<b>Low</b>	<b>High</b>

To facilitate the discussion about the two options, IT also provides an increment list, showing what each additional requirement between "The 80/20 Design" and "The Ideal Design" provides and costs in time, development, and other costs. Remember, those costs may include customization, which adds a hit to ongoing maintenance. The full costs of that final vision needs to be captured and IT does it here.



## Pareto's Principle

### Applying the 80/20 rule for project scope and success control

Now, IT works with the users and business sponsors to decide on the final scenario, or “**The Approved Design**”. IT isn't the heavy here. The IT team simply provides the costs of the design choices, and the users work with sponsors to sell the business case for the benefit and arrive at an approved and funded design. How this looks in the management summary is shown below:

Major Project Scope Drivers	80/20 Design (\$K)	Ideal Design (100% Scope) (\$K)	Approved Design (\$K)
Standard ERP Installation	\$3,000	\$3,000	\$3,000
Reporting:			
<b>Standard Reporting</b>	\$100		
<i>Custom Reporting</i>		\$500	\$500
Consolidation			
<b>Standard Financial Consolidation</b>	incl		Incl
<i>Third Party Consolidation</i>		\$1,000	
Chart of Accounts			
<b>Standard COA</b>	incl		incl
<i>Custom COA</i>		\$500	
Product Data Management			
<b>Standard BOM Management</b>	incl		
<i>Third Party Product Management</i>		\$900	\$900
Inventory Routing			
<b>Standard Inventory Management</b>	incl		incl
<i>Custom Inventory Tracking and Routing</i>		\$1,000	
Sales Commission Calculation			
<b>Interfaced commission data</b>	\$30		\$30
<i>Customized commission transactions</i>		\$300	
User Interface			
<b>Standard User Interface</b>	Incl		incl
<i>Custom User Interface</i>		\$800	
<b>Total Cost (\$K)</b>	\$3,130	\$8,000	\$4,430
<b>Duration</b>	12 months	28 months	28 months
<b>Risk</b>	Low	High	Medium



## **Pareto's Principle**

### ***Applying the 80/20 rule for project scope and success control***

---

Once "The Approved Design" is signed off by sponsors for scope and funds, the design phase ends and it's on to Development, with a very different perception and expectation set from the typical approach.

### ***Conclusion***

When IT tries to control requirements (even with the best of intentions) by trying to filter expectations we create contention with user communities that helps no one.

Instead, allowing for full requirements gathering and then applying the "80/20 rule" in the design phase frames the project with a collaborative business approach. Decisions are made by the right people based on full information. The extra effort in the design phase drastically improves probability of satisfaction and success, while driving the most business benefit from a well-managed investment plan.

### **For More Information**

For more information on this or any other Management Agility content, go to [www.managementagility.com](http://www.managementagility.com).