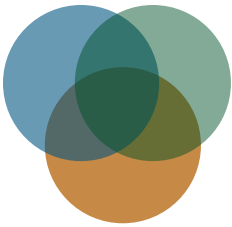


OPTIMIX PRODUCT

OVERVIEW AND OPERATING INSTRUCTION





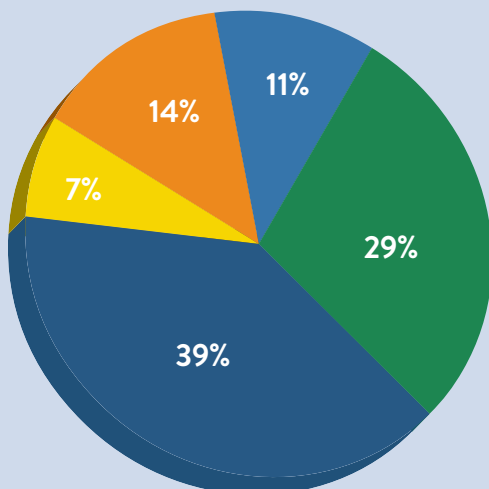
INTRODUCTION

There is never enough time or money to do everything. The scope of efforts must be contained to the available time and resources. The natural expectation is for executives and stakeholders to want it all and want it all now. Determining how much can be accomplished realistically in the available time frame and money always presents a challenge. More importantly is spending those resources on the highest value items is paramount to the success of organizations. Losing control over the effort is the first step on the road to disappointment. Managing the scope of effort is a never-ending task. Frequent and open reviews are a must to set realistic expectations and create a successful environment.

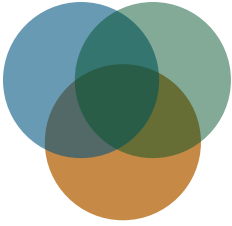
In order to assess any item such as a program, project and requirement you need to establish and have standard a protocol. Certainly alignment to corporate goals and stakeholder feedback has to be part of the protocol. These mechanisms cannot be easily quantified. However, they can influence quantified elements. The three elements that you need to consider for each item are cost, risk, and gain. By just focusing on these three elements you can greatly clarify and optimize your portfolio and ensure that you are moving in the right direction.

QUESTION ASKS IT EXECUTIVE

“What is your general process for optimizing requirements? (Pick the one closest to your method)”



Priority based on gain:	14%
Priority based on risk:	11%
Priority based on gain and risk:	29%
Priority based on stakeholder feedback:	39%
Other or Non-applicable:	7%



PORTFOLIO OPTIMIZATION AND MANAGEMENT

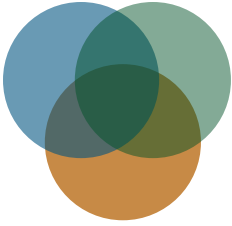
The problem with portfolio management is that it is hard and error prone. It also is not easily optimized. Calculating costs, albeit difficult, is understood. Calculating return on investment is less understood and much more difficult. It can take months to create valid ROI and over 75% of IT executives claim their organization is not good at doing it. Risk is even more important and difficult. Ninety- four percent of IT executives believes risk assessment is an important function. Yet, the majority of organizations claim they are moderately to poorly skilled. What is generally understood is the relationship between items. For example, one item being of higher value or lower value, or the same item more complex or easier.

The Standish OptiMix is a secure, 100% web-based Software-as-a-Service (SaaS) application available for deployment on-demand at a carrier-grade hosting facility. The OptiMix product has been designed to greatly simplify the ROI and Risk assessment for optimization purposes. ROI uses a five-option select box process for very high value to very low value. Risk also uses a five-option select box for very complex to very easy items. This method to set simple values makes it very easy and effective. More importantly people are more likely to enter the information because it takes so little time and is straightforward. The key to the approach is relativity and research. Relativity meaning the ROI and Risk assessments are based on the comparison to the items in the optimization case. Our balance features insures proper distribution across the case. Research means for each level for both ROI and Risk there is a corresponding default based match on our primary research.

QUESTION ASKS IT EXECUTIVE

“In your opinion, please rate and rank in importance for compliance and governance skills in implementing and maintaining general investment, TCO assessments, ROI assessments, and risk assessments.”

INVESTMENT	HIGHLY IMPORTANT	IMPORTANT	SOMEWHAT IMPORTANT	NOT IMPORTANT	RANK
General	29%	42%	28%	1%	2.7
Risk Assessment	51%	43%	6%	0%	3.9
ROI Assessment	18%	63%	19%	0%	2.8
TCO Assessment	20%	49%	31%	0%	2.7

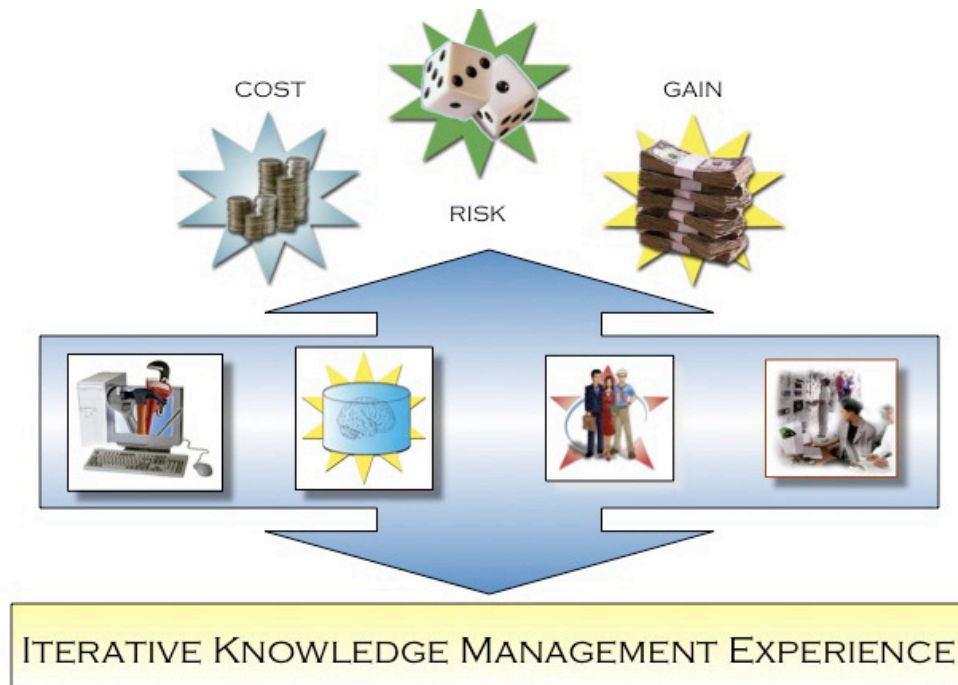


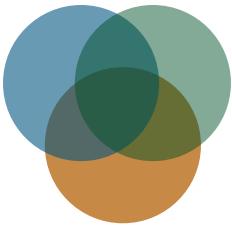
ITERATIVE EXPERIENCE

OptiMix uses the patented process of converting logical constraints into linear constraints. Such a process allows you to optimize products based on dependencies. There are two of dependencies: existence and time. Existence simply means if something exists then something else must exist or not exist. Timing means an item must be completed before or after another item. There are three optimization models: Maximize Gain in which OptiMix will present the optimal set of projects that give the organization the greatest gain for the budget regardless of risk. Minimize Risk in which OptiMix will present the optimal set of items that give the organization the least risk for the budget regardless of gain. Maximize gain with a calculated risk in which OptiMix will present the optimal set of items that give the organization the least risk and greatest gain for the budget.

OptiMix offers an iterative knowledge management experience. The experience is gained by doing alternative what-ifs. OptiMix will allow the organization to play with budgets, project, features, constraints, and other factors. OptiMix can be used to set a project in motion based on the position within the stack of items and have rapid feedback on the execution of those items. OptiMix aids in the planning and decision process of which items to budget and execute. OptiMix aids when budgets get cut or costs need to be reduced. OptiMix maximizes the value of your portfolio of investments.

CLOSED LOOP SYSTEM





PRODUCT OVERVIEW

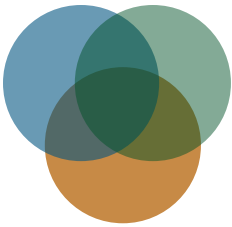
The product overview starts with the case listing page. Case listing: provides a view of all your cases, which includes: case name, case types and actions. Case name: is assigned by the user. You should create a name that will indicate the program or project. It also helps you remember the case. Our examples will be using the case name of “NewZoo”. Case type: is assigned by the user on creation, but can be changed at any time. View Results: is a variable field. In the normal position, by clicking on it you will see the current results. If you are optimizing a case, it will read either “Pending” or “Checked Out”. Pending means the case is sitting in queue prior to submission. Checked Out means the case is in the optimizer engine. Delete: means the case is totally erased and gone from the system. Copy: means the entire case is duplicated including dependencies.

Create New Case: at the top of the page is where you add a new case. Now click on that to start optimizing.

OPTIMIZER CASE LISTING

[Create New Case](#)

Case Name	Case Type	Actions
D0-Decider	Program	Edit View Results Delete Copy
D1-Issue Creation	Project	Edit View Results Delete Copy
D2-Reporting	Project	Edit View Results Delete Copy
D3-Excite	Project	Edit View Results Delete Copy
D4-Pitch	Project	Edit View Results Delete Copy
OptiMix	Program	Edit View Results Delete Copy
Z0-NewZoo	Program	Edit View Results Delete Copy
Z1-Cat Mountain	Project	Edit View Results Delete Copy
Z2-Elephant Field	Project	Edit View Results Delete Copy
Z3-Bear River	Project	Edit View Results Delete Copy
Z4-Monkey Tree	Project	Edit View Results Delete Copy
Z5-Birds	Project	Edit View Results Delete Copy
Z51-Aviary	Microproject	Edit View Results Delete Copy
Z52-Duck Pond	Microproject	Edit View Results Delete Copy
Z6-Snake Pit	Project	Edit View Results Delete Copy
Z7-Deer Run	Project	Edit View Results Delete Copy
Z8-Exotic Land	Project	Edit View Results Delete Copy
Z81-Outback	Microproject	Edit View Results Delete Copy



BUILDING A PROGRAM CASE

First, on the OptiMix home page click “Create New Case”. This brings you to a blank form. Give it a name; select “Program” as the case type, fill-in the budget amount and select a model.

Second, give each project a name, ROI rating, complexity rating, status (not mandatory, mandatory or inactive), goal rating cost (in thousands) and percent of labor. Do this for each project. In our zoo example we start with Cat Mountain with very low ROI and very easy, but mandatory, the cost \$300,000 and it is 100% labor cost. FYI: percent of labor is used to calculate risk. You add a row by clicking on the add rows button. Once you are satisfied with the case input, hit save and continue.

Case Name:

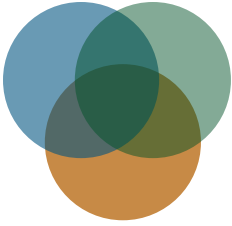
Case Type:

Total Budget (in thousands):

Optimize for:

Include in Portfolio? Yes No

Delete	Project Name	ROI	Complexity	Status	Goal	Cost (in thousands)	% Cost that is Labor Cost (nearest whole percent)
<input type="checkbox"/>	Cat Mountain	Very Low	Very Easy	Mandatory	Precise	300	100
<input type="checkbox"/>	2 Jaguars	High	Complex	Not Mandatory	Vague	350	100
<input type="checkbox"/>	2 Lions	High	Average Complexity	Mandatory	Precise	150	100
<input type="checkbox"/>	2 Panthers	Average	Average Complexity	Not Mandatory	Loose	125	100
<input type="checkbox"/>	4 Tigers	Average	Average Complexity	Not Mandatory	Precise	250	100
<input type="checkbox"/>	Elephant Field	Very Low	Very Easy	Not Mandatory	Distant	100	100
<input type="checkbox"/>	4 Elephants	High	Average Complexity	Not Mandatory	Close	240	100
<input type="checkbox"/>	4 Rhinos	Average	Average Complexity	Not Mandatory	Close	300	100
<input type="checkbox"/>	2 Giraffes	Average	Complex	Not Mandatory	Close	100	100
<input type="checkbox"/>	4 Zebras	Average	Complex	Not Mandatory	Close	60	100
<input type="checkbox"/>	Bear River	Very Low	Very Easy	Not Mandatory	Close	100	100



TYPES OF OPTIMIZATION CASES

There are four types of cases. These types refer to the relationship between programs, projects, microprojects, features, steppingstones, iterations and story cards.

PROGRAM: optimizes on a group of projects or a single program. This optimization type looks at the projects within a program so the organization can focus on the highest value projects.

PROJECT: optimizes on features or microprojects within a specific project. This optimization type looks at the features within a project so the organization can focus on the highest value features. This type also provides the ability optimize microprojects within an agile project.

MICROPROJECT: optimizes on steppingstones/iterations within microproject. For an agile project, this optimization type looks at the steppingstones/iterations within a project so the organization can focus on the highest value iterations.

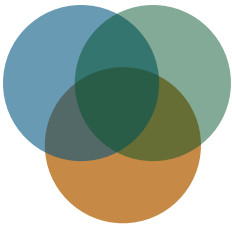
STEPPINGSTONES/ITERATIONS: optimizes on story cards for agile projects. This optimization type looks at the story cards within a microproject and lays out iteration sequence so the organization can focus on the highest value story cards.

To select an optimization type, select from the pull down as show below:

CASE LISTING

Please Note: Budget and Cost values are to be entered in the nearest whole amount. No decimal values are allowed.

	Case Name: <input type="text" value="Zoo"/>
	Case Type: <input type="text" value="Steppingstones (Optimize on the Steppingstone's story cards)"/>
	Total Story Points: <input type="text" value="500"/>
Story Points per Steppingstone/Iteration:	<input type="text" value="50"/>
	Optimize for: <input type="text" value="Minimal Risk"/>



OPTIMIZATION MODELS

There are three optimization models: maximize gain, minimize risk, and maximize gain with a calculated risk.

MAXIMIZE GAIN: optimizes items such as projects, requirements, and story cards in order of the highest payback regardless of risk. If the cost of the project is higher than the budget it will leave out the lowest gain items out of the result set. Lower gain items that are marked mandatory will be included in the results set. Lower gained items that have dependence to higher gain items will lower the gain of those items and be sequenced accordingly.

MINIMIZE RISK: optimizes items such as projects, requirements, and story cards in order of the lowest risk regardless of gain. The cost of the project must higher than the budget and system will leave out the highest risk items out of the result set. Higher risk items that are marked mandatory will be included in the results set. Higher risk items that have dependence to lower risk items will increase the risk of those items and be sequenced accordingly.

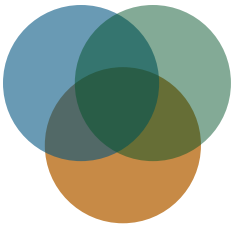
MAXIMIZE GAIN WITH CALCULATED RISK: optimizes items such as projects, requirements, and story cards in order of the highest payback with the lowest risk. If the cost of the project is higher than the budget it will leave out items with lowest gain and highest risk items out of the result set. Items that are marked mandatory will be included in the results set no matter of the how they get calculated but they will be sequenced accordingly. Dependences will affect the selection and sequence.

To select an optimization model, select from the pull down as show below:

The screenshot shows a 'CASE LISTING' form with the following fields and values:

- Case Name: Zoo
- Case Type: Steppingstones (Optimize on the Steppingstone's story cards)
- Total Story Points: 500
- Story Points per Steppingstone/Iteration: 50
- Optimize for: Minimal Risk

A blue arrow points to the 'Optimize for' dropdown menu.



ROI/VALUE

Return on Investment and value assessments can take a long time to create. In the end they are generally little more than an educated guess. Our approach is to assess value from high to low. We have collected data on projects and assigned average value weights. When considering ROI or value for the optimizer the assessment for individual item should be in relationship with the other items in the case. In other words does this project have a higher ROI than this other project? An item can have one of five types of ROI or Value ratings from very high to very low.

VERY HIGH ROI/VALUE: are items with the most value. A balanced portfolio will have 10% of very high value items.

HIGH ROI/VALUE: are items that are not most valuable, but close to it. A balanced portfolio will have 20% high value items.

AVERAGE VALUE: are items that are common. Nice to have, but may not add much value. A balanced portfolio will have 40% average value items.

LOW VALUE: are items that may be necessary to run the application or to meet compliance. By themselves they return little to nothing for the investment. A balanced portfolio will have 20% low value items.

VERY LOW VALUE: are items that also may be necessary to run the application or to meet compliance. By themselves they return little to nothing for the investment. A balanced portfolio will have 10% very low value items.

Case Name:

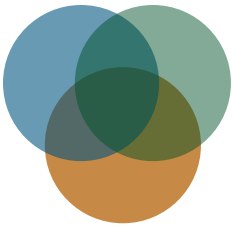
Case Type:

Total Budget (in thousands):

Optimize for:

Include in Portfolio? Yes No

Delete	Requirement Name	ROI	Complexity	Status	Goal	Cost (in thousands)	% Cost that is Labor Cost (nearest whole percent)
<input type="checkbox"/>	Cat Mountain	Very Low	Very Easy	Mandatory	Precise	300	100
<input type="checkbox"/>	2 Jaguars	High	Complex	Not Mandatory	Precise	350	100
<input type="checkbox"/>	2 Lions	High	Average Complexity	Mandatory	Precise	150	100
<input type="checkbox"/>	2 Panthers	Average	Average Complexity	Not Mandatory	Precise	125	100
<input type="checkbox"/>	4 Tigers	Average	Average Complexity	Not Mandatory	Precise	250	100



COMPLEXITY

Like, Return on Investment, complexity assessments can take a long time to create. They too in the end are generally an educated guess. Our approach to complexity is from very complex to very easy. We have collected data on projects and assigned average complexity weights. We convert complexity into risk. When considering complexity for the optimizer the assessment for an individual item should be in relationship with the other items in the case. In other words, does this project have a higher complexity than other projects? An item can have one of five types of complexity ratings from very complex to very easy.

VERY EASY: are items that have a standard activity. They are items that have been done many times before and well known to the teams. A balanced portfolio will have 10% of very easy items.

EASY: are items that have a fairly standard activity. They are items that have been done many times before and well known to the teams. A balanced portfolio will have 20% easy items.

AVERAGE COMPLEXITY: are items that are common, but may not be fully understood by the team. They may have been done before, but not often. A balanced portfolio will have 40% average complexity items.

COMPLEX: are items that are not common and their requirements are fuzzy and not understood by the team. They will not have been done before or if done before not well. A balanced portfolio will have 20% complex items.

VERY COMPLEX: are items that are ground breaking and their requirements are very vague and not understood by the team. They will not have been done before or if done before not well. A balanced portfolio will have 10% very complex items.

Case Name:

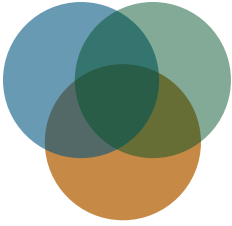
Case Type:

Total Budget (in thousands):

Optimize for:

Include in Portfolio? Yes No

Delete	Requirement Name	ROI	Complexity	Status	Goal	Cost (in thousands)	% Cost that is Labor Cost (nearest whole percent)
<input type="checkbox"/>	Cat Mountain	Very Low	Very Easy	Mandatory	Precise	300	100
<input type="checkbox"/>	2 Jaguars	High	Complex	Not Mandatory	Precise	350	100
<input type="checkbox"/>	2 Lions	High	Average Complexity	Mandatory	Precise	150	100
<input type="checkbox"/>	2 Panthers	Average	Average Complexity	Not Mandatory	Precise	125	100
<input type="checkbox"/>	4 Tigers	Average	Average Complexity	Not Mandatory	Precise	250	100



ALIGNMENT

Alignment with corporate, business unit goals should be a relatively easy task with little time to create. Like ROI and complexity, goal alignment is an educated guess. Our approach is to assess the project from precisely matching the business goal to distant from the corporate goal. However, unlike ROI and complexity, goals are neither weighted nor optimized. Goal types are displayed in the stack of projects and features as to help managers and executives make decisions on which projects should be included in the portfolio. When considering corporate goals an individual item should be in relationship with the other items in the case. In other words, is this project more precisely aligned than other projects? An item can have one of five types of goal ratings from precise to distant.

PRECISE: are items that perfectly conform to the corporate goal. For example if the goal of the company is to increase software as service revenue, than projects that advance this goal would be perfectly aligned. A balanced portfolio will have 10% of projects that are perfectly aligned.

CLOSE: are items that closely conform to the corporate goal. Again, if the project was to increase customer satisfaction within software a service, but with other services as well, this may be considered closely aligned. It would be relative to the precise projects. A balanced portfolio will have 20% close items.

LOOSE: are items that loosely conform to the corporate goal. In this case maybe it is an IT project that modernizes infrastructure that will eventually be used in the software as a service business. Remember loose may support the corporate goal, but not as precise or close as other projects in the stack. A balanced portfolio will have 40% loose items.

VAGUE: are items that may or may not conform to the corporate goal. They are however further away from the corporate goal than the other 3 types of projects. They will have sponsorship from executives. They also may be projects that are required by government regulation or have operational improvements. In this case the project might be an internal accounting or human resource system. A balanced portfolio will have 20% vague items.

DISTANT: are items that are furthest away from the corporate goal than all the other types of projects. They also may be mandatory or operational projects. A balanced portfolio should have 10% distant items.

Case Name:

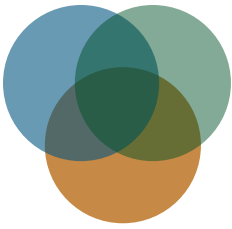
Case Type:

Total Budget (in thousands):

Optimize for:

Include in Portfolio? Yes No

Delete	Requirement Name	ROI	Complexity	Status	Goal	Cost (in thousands)	% Cost that is Labor Cost (nearest whole percent)
<input type="checkbox"/>	Cat Mountain	Very Low	Very Easy	Mandatory	Precise	300	100
<input type="checkbox"/>	2 Jaguars	High	Complex	Not Mandatory	Precise	350	100
<input type="checkbox"/>	2 Lions	High	Average Complexity	Mandatory	Precise	150	100
<input type="checkbox"/>	2 Panthers	Average	Average Complexity	Not Mandatory	Precise	125	100
<input type="checkbox"/>	4 Tigers	Average	Average Complexity	Not Mandatory	Precise	250	100

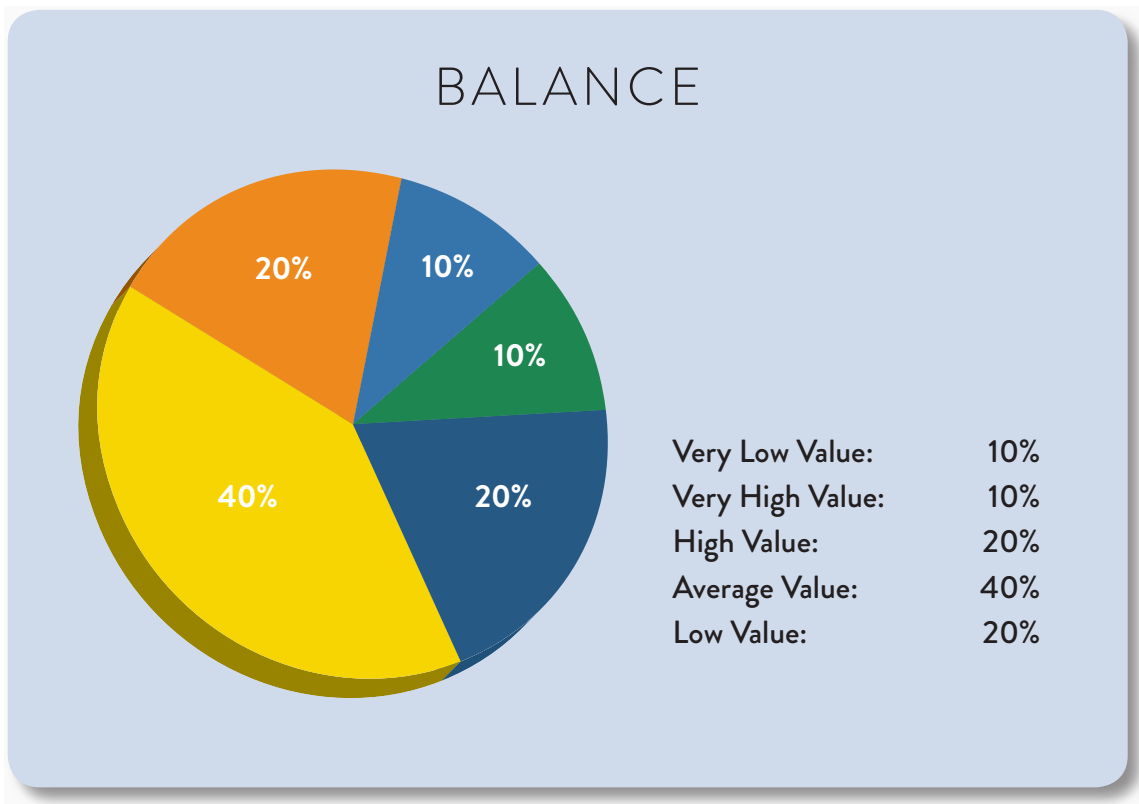


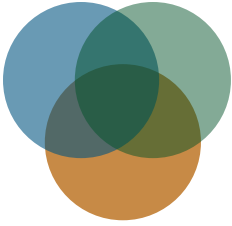
BALANCE CASE

Over any set of items there will be distribution. The first thing you should do is view your distribution by hitting the “View Balance Button”. This will provide you with a set of charts to view the distribution of your case. In our distribution we use 10% at the extreme end, 20% in the middle areas and 40% for center. This distribution is essential for a true optimization to work. To balance and rebalance your case, click balance case at the bottom of the case page. The balance process moves from the outer or extreme edge inward. For example the system will start with Very High ROI. This should be 10% of the projects in the portfolio. If there is not enough projects that are very high ROI, the system will highlight the projects that are ranked high re-rank them and suggest a number to change to equal 10%. If there are too many projects that are very high ROI, the system will highlight the projects that are ranked very high re-rank them and suggest a number to change to equal 10%.

The system will do this for ROI/Value, complexity and goals. Once you are satisfied with your balanced program, hit save. Save will save the case and bring you back to the case listing or home page. Here you might want to view the charts again, by hitting the view balance button.

Note: Hitting the balance button will also automatically save your work before you balance.





DEPENDENCIES

The main feature of the **OptiMix** Product is the ability to optimize items with constraints. Our solution is a Standish patented process to convert logical dependencies into linear constraints, and then you optimize the case. There are two types of dependencies: existence and timing.

EXISTENCE DEPENDENCY: provides a way to consider the relationship between items. For example if there are elephants in the zoo, then there must be a place to put them, such as an elephant field. On the other hand if elephants are not in the results set, you do not need the elephant field.

Enter Project Existence Dependency below.

Enter Existence Dependency

IF exists in the Optimized Case,

THEN must exist

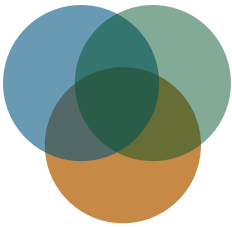
ORDERING DEPENDENCY: provides a way to consider the timing between items. For example if there are elephants to be in the zoo, then there must be a place to put them before you have elephants.

Enter Ordering Dependency

DO

BEFORE

Once all the dependencies have been entered you ready to optimize and output the results.



RESULTS

From the case listing page select the case by clicking on the edit button and then on the bottom of the case page press save and optimize. The system will take you back to the case-listing page and display the status of the optimization process from pending to checked-out to view results. Once the status is changed to view results, you can click on view to see the results and print them out if you want.

Here is what a results page looks like showing the name of the case, the total budget, the cost of projects accepted, the estimated gain (3 year ROI), the payback period in years and the amount at risk. It is good practice to keep the results cost and the calculated risk to be at or under the budget. The projects are in order of their priority based on the case model.

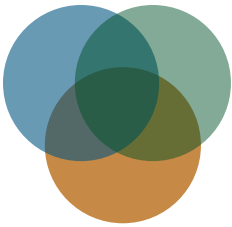
Case Name: Zoo
Total Budget (in thousands): 1,500
Optimize for: Maximum Gain
Results Cost (in thousands): 1,050
Results Gain: 494
Payback Period (Years): 2.1
Calculated Risk: 296

Results Item List

Item Name	ROI	Complexity	Status	Goal	Cost (in thousands)	% Cost that is Labor Cost (nearest whole percent)
Aviary	Very Low	Very Easy	Not Mandatory	Vague	50	100
2 Condors	Very High	Very Complex	Not Mandatory	Vague	200	100
Cat Mountain	Very Low	Very Easy	Mandatory	Precise	300	100
2 Jaguars	High	Complex	Not Mandatory	Precise	350	100
2 Lions	Average	Average Complexity	Mandatory	Precise	150	100

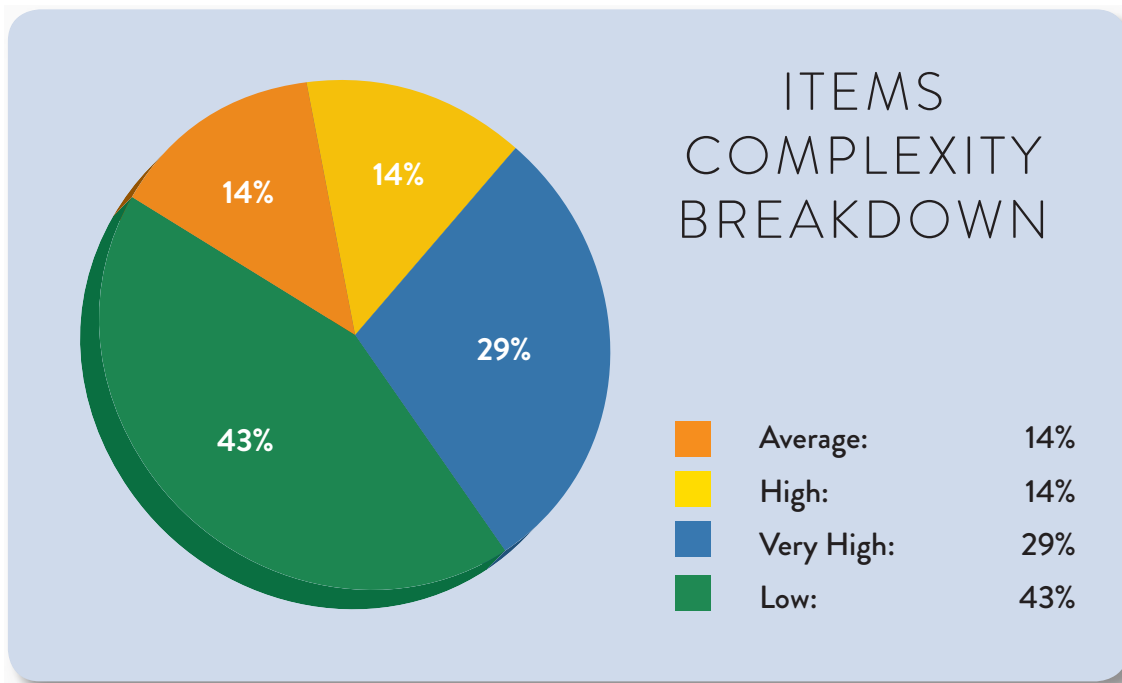
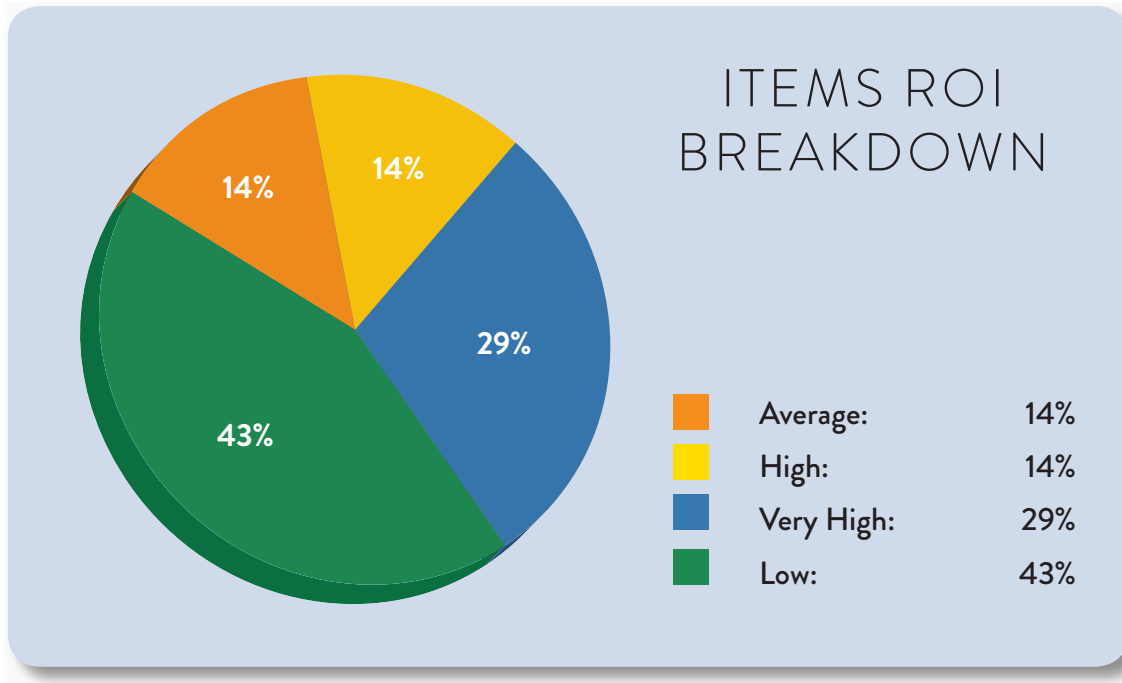
Items Not in Results List

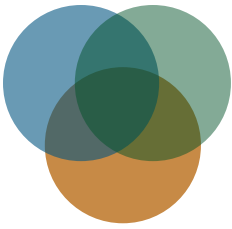
2 Panda Bears	Very High	Very Complex	Not Mandatory	Close	600	100
Wooly Mammoth	Very High	Average Complexity	Not Mandatory	Distant	400	100
2 Gorillas	Very High	Average Complexity	Not Mandatory	Loose	300	100



GRAPHIC RESULTS

From the text results page you can review the results in graphic form. Just click on view charts. Here is example of the results in pie charts.





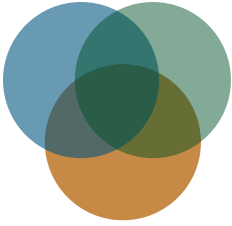
EXPERIMENTING

It is highly recommended that you go back and forth between the different models. The what-if scenarios will help to develop the optimum plan. You can make changes such as make projects as mandatory or inactive. You can change cost, budgets, ratings, and percent of labor. You can do this until you feel you have the right set of projects for your organization. You can also update the program case at anytime as either completion or cancelation resolves projects.

Here is same zoo program case, only showing minimal risk. Please note the cost is same, but the gain went from \$494,000 to \$334,000. This cause the payback period to jump by one year from 2.1 years to 3.1 years. The risk went from \$296,000 to down to \$256,000. The projects are in order of their priority based on the minimize risk model.

Case Name: Zoo
Total Budget (in thousands): 1,500
Optimize for: Minimal Risk
Results Cost (in thousands): 1,050
Results Gain: 334
Payback Period (Years): 3.1
Calculated Risk: 256

Results Item List						
Item Name	ROI	Complexity	Status	Goal	Cost (in thousands)	% Cost that is Labor Cost (nearest whole percent)
Aviary	Very Low	Very Easy	Not Mandatory	Vague	50	100
Bear River	Very Low	Very Easy	Not Mandatory	Close	100	100
Monkey Tree	Very Low	Very Easy	Not Mandatory	Loose	100	100
Cat Mountain	Very Low	Very Easy	Mandatory	Precise	300	100
2 Lions	Average	Average Complexity	Mandatory	Precise	150	100
2 Jaguars	High	Complex	Not Mandatory	Precise	350	100
Items Not in Results List						
10 Swans	Low	Easy	Not Mandatory	Loose	5	100
10 Mergansers	Low	Average Complexity	Not Mandatory	Vague	5	100
10 Goldeneyes	Low	Average Complexity	Not	Close	5	100



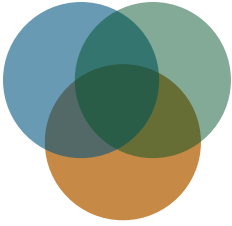
REQUIREMENTS AND AGILE

The **project and microproject models** are very similar to the program model. The project model looks at requirements instead of projects. In the background the ROI and risk calculation are different. The microproject model uses total story points instead of budget and iteration use story points as well. In addition, there is no percent of labor; however the output looks very similar to programs and projects. The big change in output is in the steppingstone model. The model asks for a story point per iteration. Then the output shows optimizing a microproject (release) by iteration. Here you would also go back and forth through the models until you received the best results for your organization. Here is an example of the zoo using steppingstones.

RESULTS FROM OPTIMIZATION

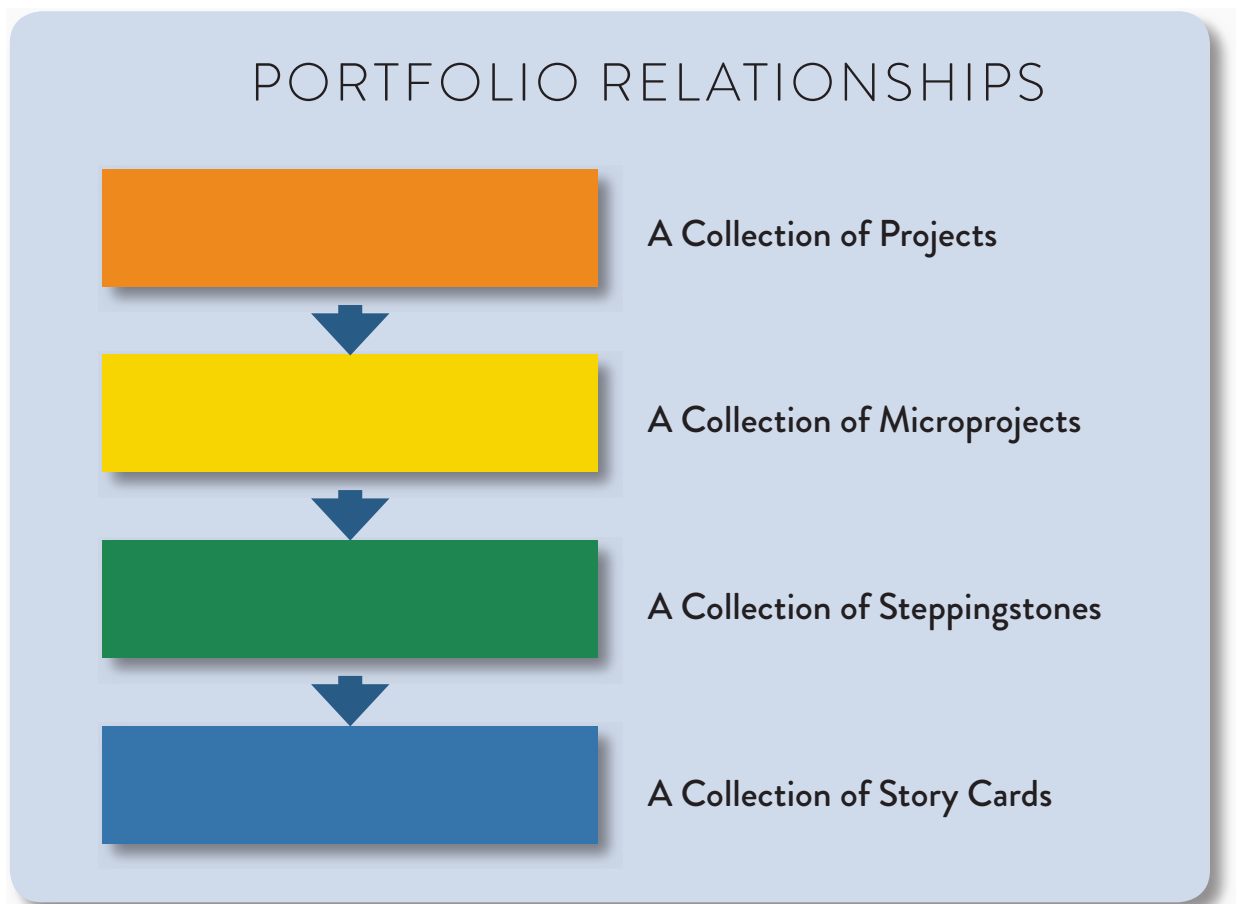
Case Name: Z821-Carnivores				
Total Story Points: 120				
Story Points per Steppingstone/Iteration: 60				
Optimize for: Maximum Gain				
Results Story Points: 120				
Results Gain: 107				
Payback Period (Years): 1.1				
Calculated Risk: 35				
Results Item List				
Item Name	Value	Complexity	Status	Story Points
Steppingstone/Iteration: 1				
T-Rex	Very High	Very Complex	Not Mandatory	50
Baby T- Rex	Very High	Very Complex	Not Mandatory	10
Steppingstone/Iteration: 2				
Acrocanthosarus	Very High	Very Complex	Not Mandatory	30
Baby Aldebertosaurus	Very High	Very Complex	Not Mandatory	5
Adasaurus	High	Complex	Not Mandatory	25
Items Not in Results List				
Aldebertosaurus	High	Complex	Not Mandatory	36
Baby Adasaurus	High	Complex	Not Mandatory	5
Alectrosaurus	Average	Complex	Not Mandatory	43
Compsognathus	Average	Complex	Not Mandatory	26

An active portfolio management system can provide the assistance necessary to frequent and transparent reviews. There are normally only two ways to manage a portfolio of effort, either top down or bottom up. In the top down approach you start with an over arching set of programs and move down through projects and requirements. In the bottom up approach you start with service requirements and move up to the portfolio level. The third way is neither. If you have the right tool you can start at the top, bottom or the middle. You also should be able to make changes iteratively and have the system automatically update the portfolio.

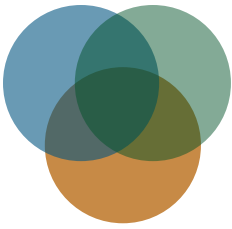


BUILDING A PORTFOLIO VIEW

OptiMix is part of an overall healthy project management ecosystem. A healthy project management ecosystem will both save and make money for your company. OptiMix (project portfolio optimizer) allows for the shift of evaluating individual and ad-hoc projects as one offs by identifying real value across a portfolio of projects. Additionally, it will save time and improve project management performance by identifying, measuring, and isolating sources of opportunity and waste. In our portfolio management optimizer system there are four basic relationships. The entire portfolio is made-up of programs, projects, microprojects and steppingstones. There is the natural relationship on programs. Programs are made up of projects. Projects can have either features or microprojects. Microprojects have steppingstones. Steppingstones have story cards.



In order to manage your project portfolio click on Portfolio View from the case listing OptiMix home page. That will bring you to the Optimizer Portfolio Summary page. Then click on the program, project or microproject that you want to manage. Here we are clicking on the Z0-NewZoo Program. This will bring you to the Optimizer Portfolio Mapping page to allow you to map projects under the D0-NewZoo program.



MAPPING YOUR PORTFOLIO

Mapping your portfolio is as easy as dragging and dropping items into place. Here we just map our available zoo projects under our D0-NewZoo program. If we want to add Z5-Birds we just drag it over. If we don't want the Monkey Tree we leave it in the available item stack. Do the same for projects and microprojects. Under your projects you move your microprojects. Under your microprojects you drag your steppingstones. Once all items have been placed you are now done. However, you can move items back and forth as you see fit to manage your portfolio. Projects that don't have a program are put under the orphan program. This is also true for orphan microprojects and steppingstones.

PORTFOLIO VIEW

Optimizer Portfolio Mapping for Program: Z0-NewZoo

Drag and Drop items from the "Available Items" container into the "Mapped Items" container to associated the item with the current the "Mapped Items" by dragging and dropping them into the "Available Items" container.

Mapped Items

Z1-Cat Mountain
Z2-Elephant Field
Z3-Bear River
Add Items Here

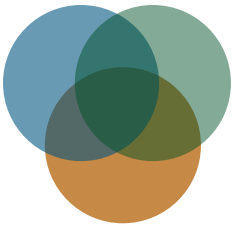
[Save](#)

Available Items

Z4-Monkey Tree
Z5-Birds
D1-Issue Creation
Z6-Snake Pit
Z7-Deer Run
Z8-Exotic Land

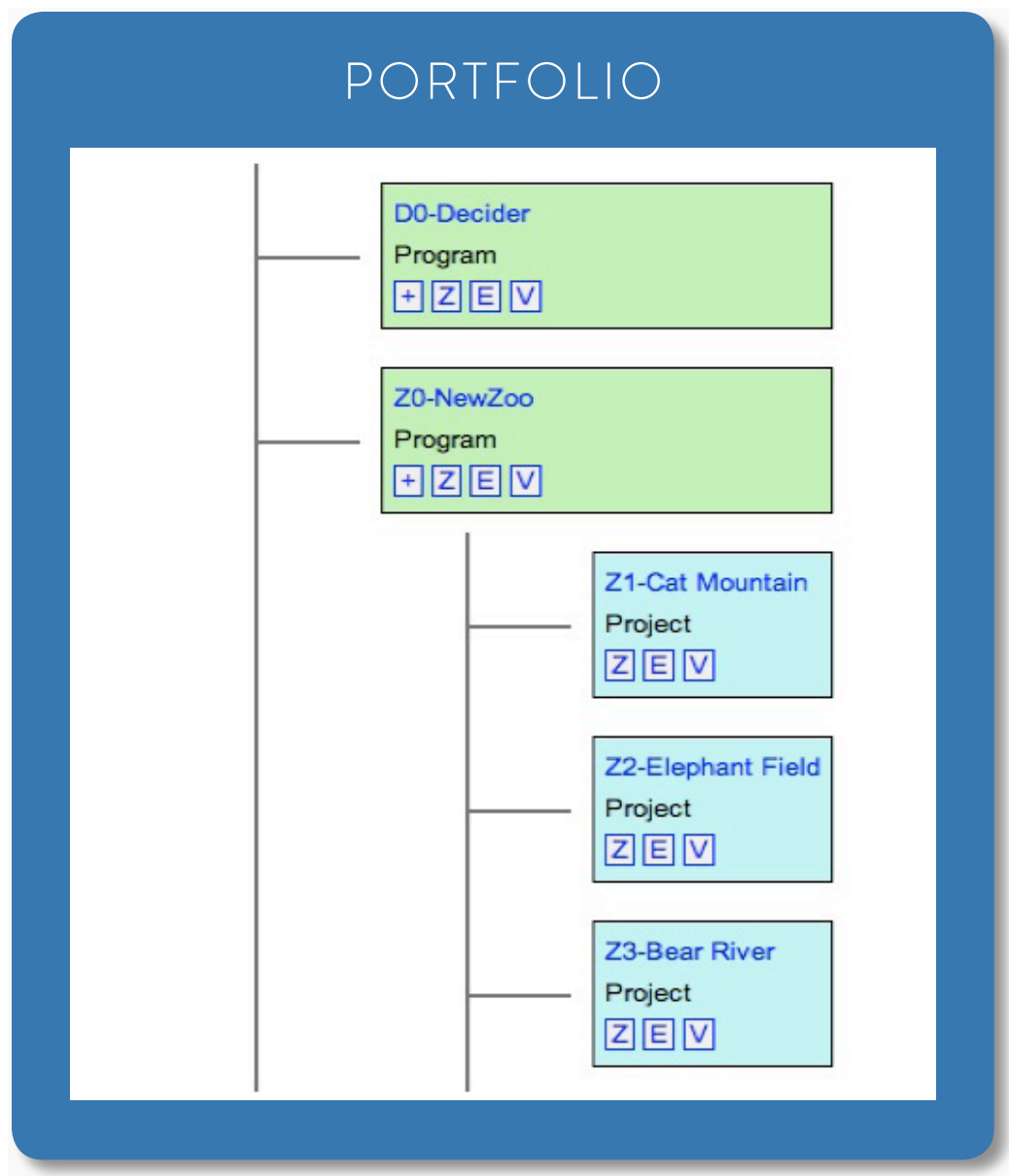
[Webmaster](#) | [Privacy Policy](#)

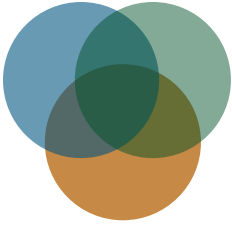
All contents are Copyright © 2010 by The Standish Group International, Inc.



MANAGING YOUR PORTFOLIO

You can view you a portfolio two ways, as an organization chart or in a summary report. To view your portfolio in the organizational chart view click on View Org Chart from the summary page. Here you will see you portfolio in boxes starting from programs to steppingstones. The box will first have the case name in blue, then it will have the type in black such as program, project, etc. Underneath the type there will be 3 to 4 boxes. If there is a plus sign it means that this is a parent item and there are children attached to the parent. Z allows you to zoom on the item. E allows to you edit the case. V allows you to view the case. If you hover over the type (program, project, etc.) it will display the details of the case.





OPTIMIZING YOUR PORTFOLIO

The Optimizer Portfolio Summary page has a lot of good detail information. For example you can look at projects with a program to see how close the total of the projects match the total cost in the program. This is the same for microprojects and steppingstones.

CASE LISTINGS

Optimizer Portfolio Summary

Click on a Case name to go to the Mapping page to start building your Org Chart.

View Org Chart

Case Name	Budget	Optimize for	Results Cost	Results Gain	Calculated Risk	Payback Period (Years)
Programs						
<input type="checkbox"/> D0-Decider	1,200	Maximum Gain with Calculated Risk	1,200	732	300	1.6
D1-Issue Creation	400	Maximum Gain with Calculated Risk	400	262	113	1.5
D2-Reporting	400	Maximum Gain with Calculated Risk	380	356	110	1.1
D3-Excite	400	Maximum Gain with Calculated Risk	340	260	92	1.3
D4-Pitch	400	Maximum Gain	10	5	2	2.0
Totals:	1,600		1,130	883	317	Average: 1.5
<input type="checkbox"/> Z0-NewZoo	4,000	Maximum Gain	3,100	2,164	965	1.4
Totals:	5,200		4,300	2,896	1,265	Average: 1.5
Projects						
D1-Issue Creation	400	Maximum Gain with Calculated Risk	400	262	113	1.5