

# How to get there

OK, so your organization has held the appropriate workshops to articulate the vision, and now everyone is wondering, “How do we get there?”. I’ve worked with many organizations that had the most detailed vision that fully identified characteristics of the end state along with the overall objectives of the program, but they struggled to turn that vision (*strategic*) into action (*tactical activities*). It wasn’t because they didn’t have the talent or skills, but because rolling out a GRC program is something that is outside of their experience and comfort zone.

I would also argue that the overall vision should be broken down to sub-strategies (depending on the size of the program) that support the grand vision. These sub-strategies are what I refer to as “streams”, with each stream being a logical grouping of program components in the areas of:

- **Use Cases**
- **Policy and/or Governance**
- **People**
- **Technology**

Keeping the number of streams to a low number (ideally under six) helps maintain alignment throughout each phase of development and implementation, and therefore inherently reduces the program delivery risk.

## The complete roadmap view

If you have ever used a GPS while driving from point A to point B, you can appreciate that looking at only the next turn directly ahead of you may get you there, but you don’t get a sense of overall progress, or your relative position to other things around you. By zooming out to get the rest of the map in view, you quickly appreciate where you really are in your trip. The same thing is true for using an overall roadmap for deploying a risk management program. Zoom out a bit and you’ll better understand the scale, complexity, duration, participation and budget.

By its very nature, a GRC program is an aggregator of other systems and data. It would not have nearly as much value if it was a stand-alone solution performing all the functions, mainly due to the fact that many of the functions it needs to gather data are already systems operating through your organization. A GRC program is an integrated toolset that brings information, processes, and resources together to provide an aggregated view of all these things, and ultimately helps management make better decisions. It adds transparency and traceability to instill confidence from management and regulators. That is good for business.



By **Kirk Hogan**  
COO, Iceberg  
[khogan@icebergnetworks.com](mailto:khogan@icebergnetworks.com)

## CHAPTER 3: HOW TO GET THERE

There are many roles, across many operational groups, that will appreciate a complete roadmap view. The Program Management Office (PMO) will undoubtedly have more confidence, as the roadmap will speak to the integrated view of all tasks needed to successfully deliver a program on time and budget. The Chief Financial Officer (CFO) will have a better understanding of the resource requirements over time by stream and phase. This view will help them defer costs until they are absolutely needed, but more importantly get a view of Total Cost of Ownership (TCO).

The real magic happens when all the participants of the GRC program can see and appreciate their role in actualizing the project. An aggregated view using the four streams I mentioned above will bring technology groups together with operations, and business groups that will ultimately become users of the solution. It also ensures that the proper governance is applied to each aspect of the complete program so that when the solution is put into service, there will be a clear understanding of roles and responsibilities to ensure deployable success.

The real magic happens when all the participants of the GRC program can see and appreciate their role in actualizing the project.

### Attribution

The concept of attribution can be complex if you really dig into it, but I'd rather err on the side of simplicity. I describe attribution as *"the ability to link something to the objective it supports"*. This means that if I cannot describe how any activity on the roadmap is somehow contributing to the realization of an objective, then I can do one of two things: do a better job of describing its connection; or remove the activity.

We will also come back to this concept of attribution in a future chapter when we discuss measuring value, since we should also be able to attribute an increase in value to a specific thing or set of things. For now however, we will simply need to identify which of the streams an activity or component supports.

### Prioritizing Activities - "HVA" or "LVA?"

The **High Value Activity (HVA)** or **Lower Value Activity (LVA)** are concepts that the personal development industry has used for years, but I've adopted them with open arms and propose they are also perfectly suited for program management.

The HVA is fairly self-explanatory, but in the spirit of completeness, I would describe an HVA as *"any activity that has an obvious and direct attribution to increasing value of the larger objectives"*. An example might be performing incremental backups on a critical information system. By comparison, I refer to LVAs as "Lower Value" and not "Low Value" on purpose. An example of a LVA might be spending a week changing formatting in an administrative manual that might get used twice a year by a single person. I believe that most activities that are placed on a roadmap or program plan have some level of value, but perhaps their attribution is not as clear as the HVAs, or the degree of improvement is in question. Regardless, having this concept available makes the conversations easier to have when trying to make prioritization calls on what activities trump other activities.

## CHAPTER 3: HOW TO GET THERE

That's not to say you couldn't define Medium Value Activities (MVAs) and No Value Activities (NVAs), but I don't think it's necessary. MVAs would automatically become HVAs once all existing HVAs were finished, and NVAs would get removed from the plan once it was confirmed they were not attributable to increasing value.

### Work packages

We've identified four streams within a GRC roadmap. If you're familiar with rolling out new solutions, you'll also know that every program goes through distinct phases. These phases will vary depending on what methodology is used, but I would suspect that most programs will either use a Waterfall methodology, or a hybrid of a Waterfall and some other type (perhaps Agile or Extreme).

Assuming this is true, it should also then be possible to group components and activities within each stream to provide modular value, starting with foundation items and evolving to those components or activities which rely on foundation items. This will be particularly helpful when you try to deliver quick wins (discussed later in this series in Chapter 6) to demonstrate incremental value of the program instead of waiting for the end state to be achieved.

Each work package should be accompanied by a Business Requirements Document, Design Specifications, Test Plans/Cases, and other regular project management artefacts. Each work package could be delivered independently, assuming that any work package inter-dependencies have been identified, and a sequence applied.

The power of the work package is that it lets you define the entire program and its components, and then negotiate each one into a specific work package to meet internal pressures for release dates, program features, or other defined milestones.

### Treating GRC as a program

If there is any one lesson we have taken away from countless GRC program deployments, is that they are just that – *programs*. Our experience has shown that once a GRC program is treated like a project with a start and an end, the chance of success or prolonged success is greatly diminished.

There is definitely a start to the GRC program, but the key difference is that there really is no end, it just becomes a part of your organization's operational evolution. More and more use cases can be supported on the same solution base, delivering more value into the organization.

Similar to a Business Intelligence Program, a successful GRC program delivers "*Risk Intelligence*," allowing executives to make decisions that are risk-based and attributable to traceable data and information sources.

### Next in this series

Chapter 4 is "*What First*", offering approaches to aligning technology, management and staff to decide on program priorities. More chapters from this series are available at [icebergnetworks.com/risk-intelligence/](http://icebergnetworks.com/risk-intelligence/)

GRC programs become a part of operational evolution and more and more use cases are supported on the same solution base, delivering more value into the organization.