

# Is Agile the Answer for Central Government Software Projects?

In the past couple of years, DCG has worked with different departments and branches of the U.S. Federal and U.K. governments to provide Agile training or to set up Agile projects. In our experience, both governments are attracted to the idea of Agile as an approach that quickly and continuously delivers value.

However, both governments have had their fair share of software project disasters over the years. So, this begs the question, is Agile the answer for central government software projects?

As always, there are pros and cons to consider:

Is Agile the Answer for Central Government Software Projects?	
Pros (if properly implemented):	Cons:
<ul style="list-style-type: none"><li>• Progress is highly visible for all stakeholders – this is particularly helpful for auditors.</li><li>• There is a focus on delivering value to the end user first.</li><li>• There is a focus on tackling the most difficult tasks in a project first.</li><li>• Customer satisfaction is prioritized over productivity.</li><li>• There is less emphasis on establishing perfect requirements in advance, and more emphasis on adapting plans to include external change and emergent capabilities.</li></ul>	<ul style="list-style-type: none"><li>• It is possible to have too many stakeholders, especially when there is no one person with clear authority to make decisions on what is acceptable.</li><li>• The current regulations around procurement (designed, to some extent, to try to avoid previous failures in waterfall projects) constrain flexibility in contracting.</li><li>• It is difficult to subcontract Agile.</li><li>• Employees are justifiably risk-averse when they are spending public money. And if something goes wrong, there is often finger-pointing.</li><li>• Current audit practices sometimes need to change to reflect the different tempo of Agile projects.</li></ul>

Our recent experiences with Agile have highlighted yet another new challenge for government implementations: timing the transition to Agile. At a time when all new government software projects are important (or they wouldn't be funded) and under close scrutiny (with a strong adversarial climate amongst the politicians), we are seeing a tendency towards what we might call “minimal impact” Agile introductions. These can take several different forms, but the most common is to introduce some Agile buzz words (and associated activities) into what is otherwise a waterfall project.

This is partly due to the fact that many government departments subcontract most or all of their software development. The path of least resistance (and the most sound option, from a regulatory perspective) is to use existing contract formats for Agile engagements. Of course, most existing government contract formats are designed for waterfall approaches – that is, delivery of precisely what is specified in advance to be the deliverable. Vendors, often accustomed to government officials arguing about contract deliverables after the fact, actually prefer these existing waterfall-based contracts because there is some protection for them in establishing clearly specified, mutually-agreed-upon deliverables in advance.

So, what happens in many Agile projects is that the vendors agree to “operate in an Agile way,” or “comply with Agile process requirements” or something similar, while still utilizing a waterfall approach. Such commitments are rarely adequately defined and rarely binding. The “what” tends to take priority over the “how” when push comes to shove, thereby negating the principles of Agile.

The final straw in failing to hold vendors accountable for commitments to Agile tends to come from the payment process included in the contract, which may not be tailored to Agile deliverables (or at least not if the vendor can avoid it).

There is hope: Both the U.S. and U.K. governments are starting to issue guidelines for contracting for Agile using actual Agile contract templates (there is an [example on our website](#)).

Of course, the tension commonly noted between the PMO and the Agile team also comes into play here. This is a particular problem when introducing Agile into projects because the necessary step for success, redefining the project governance responsibilities, is rarely done up front; instead, the need for this emerges as communication lines between the internal Agile team, the internal PMO and the vendor quickly become tangled and generate bad feelings – not a great atmosphere for designing good governance.

While this analysis may imply that Agile is doomed to fail within the government, that is not necessarily the case. We suggest that while there are a number of barriers that must be overcome to properly implement and find success with Agile within the government, it is possible, particularly with effective coaching. We believe that Agile is the answer for central government software projects, but only if sound Agile-focused contracts and clear governance are in place.