Reviewing the MQ RFE (Request for Enhancements) process

Matthew Whitehead
Published on 21/10/2019*/ Updated on 13/11/2019*

Introduction

The [RFE (Request for Enhancement)](https://integration-development.ideas.aha.io/?project=MESNS) process has been around for over 7 years and allows products like MQ to be more directly shaped by its users. In that time MQ has delivered features and enhancements to satisfy over 300 RFEs raised by our users and business partners. That’s almost one a week!

We don’t just look at individual RFEs in isolation, we also identify themes and trends based on the types of RFE we see being raised over time and this helps us shape our future plans and deliveries.

While some of you may be familiar with how the RFE process works some have observed that the RFE process could be more transparent. We thought it would be useful to explain the different stages an RFE goes through after being raised.

How does the process work today?

Most MQ RFEs end up being categorised as one of the following:

1. Features that we think are valid, useful, and would benefit a broad range of users
	* These will typically be moved into “Uncommitted Candidate” state, and at some point may be moved into “Planned for Future Release” and/or “Delivered” state once they have been implemented.
2. Features that we think are valid and useful but would benefit a smaller set of users
	* Again, these will typically be moved into “Uncommitted Candidate” state but because of the sheer number of them, very few of them are likely to be moved out of that state.
3. Features that are invalid for some reason. This could be because they relate to a product defect and should be raised as a support case. It could also be because there are alternative ways to resolve the issue using existing features that we would prefer users to use, or simply that the user is trying to achieve something that the product isn’t really designed to do.
	* Most RFEs in this category will be moved into either “Declined” or “Is a Defect” state.

(The complete list of RFE states are listed on [the RFE support page)](https://integration-development.ideas.aha.io/?project=MESNS)

The diagram below shows the states an MQ RFE typically goes through:

The process RFEs go through after they have been submitted

As the diagram shows, once a new RFE is opened it is initially reviewed by a group of senior engineers in the MQ department. This is where we identify any RFEs that we think should be addressed in another way – perhaps as a support case – and if so move the RFE into declined state. We also use it as an opportunity to highlight features that the raiser may not be aware of. Perhaps a recent addition to the product has already addressed the particular enhancement, or that alternative mechanisms are available that we recommend using.

If an RFE is valid and is something we might consider doing in the future it is assigned to the relevant area of product development. Initially it will remain in Under Consideration state but it is then the responsibility of an engineer or architect to look at in more detail, understand what an implementation of the enhancement might involve, and prioritise against other commitments.

At some point the RFE owner will move the RFE into one of the following states:

* Uncommitted candidate – we will consider delivering the enhancement in a future release of MQ
* Delivered – the enhancement has been delivered into a release of MQ
* Declined – on further investigation there is some reason why the enhancement cannot be delivered

Once an RFE has been categorised we do periodically step back and look at the existing RFEs to see how voting has progressed or which RFEs might fit into our existing strategy, and we use this to adjust our plans.

Reviewing how we work

Our tendency has been to keep open all of the RFEs that we *could* deliver, even if we’re not sure when we will get to them. Conversations with users and business partners have shown us that this isn’t always the most helpful way to deal with them.

Some users have commented that, as is described above, some RFEs sit in the same state for a long period of time. In the worst cases some users have come to the conclusion that they don’t see the benefit in raising new RFEs.

Obviously the perfect solution would be for us to implement every RFE that gets raised and have almost none waiting to be addressed. Unfortunately the sheer number and variety of RFEs that are raised every month makes this implausible. It is inevitable that with such a broad and active user base, our users are able to think of improvements or enhancements much more quickly than we can implement them.

However, what we can do is make the process more transparent, give more feedback more often, and be clearer about which RFEs fit into the 2nd category above. Those users we have spoken to have said that if their RFE will not be delivered in the next year or two they would rather be told that, rather than be left wondering if and when we will get to it. Knowing that we can’t deliver an RFE can help them with their own planning and initiate more discussions with the MQ team about longer-term workarounds. We would like to move to a situation where we do this more often, so we are as clear as possible about the likelihood of an RFE being resolved in an upcoming release.

What will the process look like in the future?

The biggest difference for MQ users is that we will now use “Declined” to mean either of the following:

1. The RFE is invalid for one of the reasons stated above
2. **The RFE is valid but doesn’t fit into our plans over the next couple of years**

The second line is new to our revised RFE process. It gives a user a clear indication that, while the idea is a reasonable one, it does not fit into our product plans in the next 1-2 years.

With these changes in place the process now looks like this (**orange blocks are new**):

The revised process RFEs go through after they have been submitted

As is the case today, RFEs won’t simply be moved into “Declined” state with no further explanation as to our reasoning. We will always try to give a clear indication of the reasons why we have declined it so that you’re able to make a decision about whether to re-open it in the future.

Will these changes affect existing RFEs as well as new ones?

MQ currently has several hundred open RFEs and it will take us a long time to apply our new process to all of them in the near future. What we would like to do is apply the new process to all new RFEs, and then slowly apply them retrospectively to existing ones.

This may mean that an RFE you raised some time ago that we moved into “Uncommitted Candidate”, may now be moved into “Declined” state.

What if I disagree with a decision you make about an RFE?

The RFE process has always allowed users to re-open RFEs 12 months after they have been declined. The changes above won’t affect your ability to do that with any RFEs we moved into “Declined” state and we encourage you to do that if your original requirement is still important to you at that time, as that helps us evaluate the ongoing need. It’s worth remembering that users can continue to vote for declined RFEs. If a declined RFE gains a large number of votes after being closed they will be taken into account if the RFE is ever re-opened.

Should I bother raising new RFEs?

Absolutely. RFEs help us to ensure that the product continues to work well for the evolving needs of administrators and developers. Every release of MQ contains features that satisfy user RFEs. Even when they can’t be delivered exactly as requested they can help shape the direction of the product and the design of related features. They also give us a way to identify alternative approaches you may not have considered or weren’t aware of. In some cases we get RFEs that would lead the user towards an anti-pattern or an architecture that we don’t typically recommend and our RFE response can suggest better practice alternatives.

**TAGS**[**IBM MQ**](https://developer.ibm.com/messaging/tag/ibm-mq/)**,**[**MQ**](https://developer.ibm.com/messaging/tag/mq/)**,**[**REQUIREMENTS**](https://developer.ibm.com/messaging/tag/requirements/)**,**[**RFE**](https://developer.ibm.com/messaging/tag/rfe/)

*by*[*Matthew Whitehead*](https://developer.ibm.com/messaging/author/mwhitehead/)

* [Twitter](https://www.twitter.com/matthew101)

4 comments on"Reviewing the MQ RFE (Request for Enhancements) process"

1. **David Awerbuch** [November 05, 2019](https://developer.ibm.com/messaging/2019/10/21/reviewing-the-mq-rfe-request-for-enhancements-process/#comment-46110)

I have to agree with Carolyn and Peter as well. “The RFE is valid but doesn’t fit into our plans over the next couple of years” – I suggest an RFE that meets this qualification be moved to ‘Archived’ or ‘Long-Term’ status, rather than ‘Declined’. ‘Declined’ state, by the definition of the word, implies it won’t be done; another state for such an RFE implies it has not been rejected, just put on a back burner so to speak. This leaves the ‘Declined’ bucket as exactly that – the RFE will not be considered further.

[Reply](https://developer.ibm.com/messaging/2019/10/21/reviewing-the-mq-rfe-request-for-enhancements-process/#comment-46110) [(Edit)](https://developer.ibm.com/messaging/wp-admin/comment.php?action=editcomment&c=46110)

1. **Matthew Whitehead** [October 25, 2019](https://developer.ibm.com/messaging/2019/10/21/reviewing-the-mq-rfe-request-for-enhancements-process/#comment-46034)

Thanks both of you for your comments. We had a lot of discussion about exactly this issue when we were considering how best to improve the process. Ultimately our priority is to give clearer guidance as to whether or not a particular RFE is likely to be delivered in the product, so that users can better plan for either scenario.

Before these changes we were moving the vast majority of valid RFEs into “Uncommitted Candidate” but we didn’t have a way of indicating which of those RFEs fitted into our near/medium term plans. Users were feeding back to us that they had RFEs sat in that state for a long time and had no indication of whether they might be delivered. We hope that by communicating more clearly which RFEs we’re unable to deliver in the near/medium term, we are helping users with their planning.

I agree that the term “Declined” used by the IBM RFE process is a little unfortunate since some users will see it is a flat out rejection of their request. Ultimately we have to stick to the defined RFE states but we will try to feed that concern back to the RFE process team. It’s really important however to reiterate that declined RFEs don’t get removed from the system and certainly don’t get forgotten about. We still look at them and users can still see and vote on them. If they are re-opened after the 12 months we will go back through the review process afresh.

[Reply](https://developer.ibm.com/messaging/2019/10/21/reviewing-the-mq-rfe-request-for-enhancements-process/#comment-46034) [(Edit)](https://developer.ibm.com/messaging/wp-admin/comment.php?action=editcomment&c=46034)

1. **PPotkay** [October 25, 2019](https://developer.ibm.com/messaging/2019/10/21/reviewing-the-mq-rfe-request-for-enhancements-process/#comment-46028)

I agree with Carolyn Elkins. I get the sense that the vast majority of RFEs will now end up “Declined”. Whether its a great idea that will eventually be seriously considered/implemented given enough time and/or additional votes, or if its the dumbest idea ever, they all get tagged as “Declined”. I’d rather see a new category for these Long Term Consideration RFEs, not lumping them all into Declined.

[Reply](https://developer.ibm.com/messaging/2019/10/21/reviewing-the-mq-rfe-request-for-enhancements-process/#comment-46028) [(Edit)](https://developer.ibm.com/messaging/wp-admin/comment.php?action=editcomment&c=46028)

1. **Carolyn Elkins** [October 24, 2019](https://developer.ibm.com/messaging/2019/10/21/reviewing-the-mq-rfe-request-for-enhancements-process/#comment-46016)

Instead of over using the ‘decline’ you should add a new classification indicting that the RFE will be revisited as conditions change. When something is declined people feel there is no point in commenting or voting for something they now realize they need (and perhaps urgently by the time they realize).