Summary:

On the agenda for the next GitLab key review is a proposal to break up this meeting into four department key reviews, and to avoid adding three net new meetings to stakeholders counters. I don't know if you've heard of the proposal to cut the number of staff in the R&D; department by half, which would be a reduction of about 50%. On the last call, you were asked about the difference between water MRs from the company and those from the community, and you responded by saying that the company only counts community contributions. I think there's a problem with R&D; wider MR rate, which is this thing doesn't really move because it's a rate, so it feels like the way to drive this up is to specifically drive community authors to contribute more than one MR per month. A lot of you have been asking questions about how we measure the number of contributions to GitLab from the community and also from external members of the GitLab team. I've been trying to come up with a way to talk about the different parts of the game that we do and how we can improve them. It's a good question and I'm glad you're on the call because I'm a big fan of community MRs and I love that you're working with Max to make that happen. On to seven, I wanted to just touch on the postgres replication issue that we've been having with the data team, which has been an issue for a while. I did ask a question on the database side and I did talk to Craig gums a little bit as well and I'm still trying to figure out, you know, if it's truly just dedicated computation old sort of resource a server. There are a number of things that we're going to do to improve the performance of the database layer at Salesforce.com, but they're not all the same things. I'm going to put into the infricky review for next week, but I'll probably going to squeeze the balloon on the next area. I just wanted to ask a couple of quick questions on the MRs in the back end of the business, number A on the age of the patients, number B on the backlog and number C on S2. I don't know if you're aware of it, but last quarter we did a key review of our code base to make sure that we weren't creating too many new bugs. I think we're doing it the wrong way if we want to measure the age of open bugs, we should be looking at the entire population that is within the SLO time. I want to go through all the key meeting metrics and look at some of the things that we should be watching out for in the coming months. There was a slight improvement in the rate of decline that we saw in the first quarter of the year, but it was still a pretty big decline. On the security front, we have seen a significant increase in the number of incidents over the last 18 months, and we have had 10 incidents in the last three months. Yes, Chris, thanks for the question, and I'll give you a little bit of context on what's going on with the MR rate. Christopher Doen, chief executive of Doen & Co, has said it's not necessary to keep raising the MR rate, because it's already high enough. Ten is a great number to hold on to, because you're more productive, so you can fix more things, so it's not necessarily opposite.

Key Points:

On the agenda for the next GitLab key review is a proposal to break up this meeting into four department key reviews. On the security front, we have seen a significant increase in the number of incidents over the last 18 months. There are a number of things that we're going to do to improve the performance of the database layer.

Decisions Made:

On the agenda for the next GitLab key review is a proposal to break up this meeting into four department key reviews. On the security front, we have seen a significant increase in the number of incidents over the last 18 months. There are a number of things that we're going to do to improve the performance of the database layer.

Action Items:

On the agenda for the next GitLab key review is a proposal to break up this meeting into four department key reviews. On the security front, we have seen a significant increase in the number of incidents over the last 18 months. There are a number of things that we're going to do to improve the performance of the database layer.