because of the 767 byte limit on the index key size.ĊWhen working with partial indexes it can be helpful to know exactly how much of the column is covered uniquely by an index of a given size. Fernando Ipar has a pretty nifty little SQL query that will give you a rudimentary peek into how well a partial index will perform. The query will tell you what percentage of rows are uniquely identified by the index. You can check out his blog post about it over here. Here is the general form of the query:Ċ– SELECT COUNT(DISTNICT(SUBSTR(<column>,1,<partial index length>))) / COUNT(DISTINCT(<column>)) * 100 FROM <table>; SELECT COUNT(DISTNICT(SUBSTR(name,1,10))) / COUNT(DISTINCT(name)) * 100 FROM customers;ĊA Little ProblemĊWith all the goodness that partial indexes offer, I have found at least one draw back. It seems that partial indexes cannot be used with aggregation functions likeĊGROUP BY. Even if the partial index does not uniquely identify each row in the table, one would think that MySQL would be able to use the partial index to at least help theĊUpdate (11/8/2011): Someone posted an interesting answer