

Eigen::internal::pgather  
< bfloat16, Packet8bf >

Eigen::internal::pgather  
< short int, Packet8s >

Eigen::internal::pgather  
< unsigned short int, Packet8us >

Eigen::internal::pgather  
\_size8

```
graph LR; A["Eigen::internal::pgather< bfloat16, Packet8bf >"] --> D["Eigen::internal::pgather_size8"]; B["Eigen::internal::pgather< short int, Packet8s >"] --> D; C["Eigen::internal::pgather< unsigned short int, Packet8us >"] --> D;
```

The diagram illustrates a specialization of the Eigen::internal::pgather template. On the left, three boxes represent different template instantiations: one for bfloat16 with Packet8bf, one for short int with Packet8s, and one for unsigned short int with Packet8us. Blue arrows from each of these boxes point to a single box on the right, which represents the common specialization Eigen::internal::pgather\_size8. This indicates that these three different template arguments all resolve to the same implementation.