Services Interaction

What options do we have to establish such communication?

a. REST apis
b. gRPC or RPC
c. Message Queues like amazon-sqs, rabbitMQ, etc.
d. Apache Kafka

For each option describe what are the pros and cons of this solution?

a. REST apis:
    i. Its well known and a lot of development is already done in this area
    ii. Every language or framework has utilities for HTTP and JSON encode/decode
    iii. It is very easy to setup and use.
    iv. Its not very fast and performant compared to other options in this list
    v. sessions can't be maintained in web services. So, each API call needs to be completely independent
    vi. we can only perform a request and get a response, so server cannot send response on its own.
b. gRPC:
    i. gRPC supports data streaming from the server along with the traditional request-response
    ii. Its extremely fast compared to HTTP
    iii. The messages are lightweight
    iv. It has code generation tool in most languages
    v. It uses protobufs to transfer data, which are a fairly new thing and requires some dedicated learning
    vi. There are not much tools and resources available on this topic. There has not been much development on gRPC outside google, so its lacks maturity.
    vii. While transferring data, it is converted to binary which is not human readable
c. rabbitMQ:
    i. Allows streaming data
    ii. It is lightweight and easy to setup
    iii. Most of the projects like rabbitMQ are open source so they are free and accessible to everyone
    iv. Hugely scalable (up to one million messages in one second in rabbitMQ)
    v. Re-reading of messages in not allowed
    vi. Ordering and partitioning of messages is there but its not as efficient as it is in kafka
d. Apache kafka
    i. Apache kafka is much more than just pub-sub system
    ii. Its has high-performance data pipelines, streaming analytics, data integration along with event driven pub-sub
    iii. Partitioning and ordering of messages is best
    iv. It has high throughput, its fault tolerant, highly scalable and available
    v. It has steep learning curve as there are alot of things to learn

vi. It is not lightweight and takes a lot of time and efforts to integrate into the application
vii. The official library is only available in java, there are open source libraries in other languages though

For each option describe what are the cases the solution fits best?

a. REST apis are best when the performance requirements are less and development needs to be fast
b. gRPC can be used when you have an experienced developer in this field
c. RabbitMQ can be used for fast message streamings. It does not have strong coupling between services, so it can be useful when the data type is not certain.
d. Apache kafka is huge. So when you have exhausted all above options and still need better performance and ready to invest time and efforts into something new, apache kafka is the best choice.