

-HTTP-Baggage

baggage

baggage

<https://github.com/gogf/gf/tree/master/example/trace/http>

```
package main

import (
    "github.com/gogf/gf/contrib/trace/otlphttp/v2"
    "github.com/gogf/gf/v2/frame/g"
    "github.com/gogf/gf/v2/net/gtrace"
    "github.com/gogf/gf/v2/os/gctx"
)

const (
    serviceName = "otlp-http-client"
    endpoint    = "tracing-analysis-dc-hz.aliyuncs.com"
    path        = "adapt_*****_*****/api/otlp/traces"
)

func main() {
    var ctx = gctx.New()
    shutdown, err := otlphttp.Init(serviceName, endpoint, path)
    if err != nil {
        g.Log().Fatal(ctx, err)
    }
    defer shutdown()

    StartRequests()
}

func StartRequests() {
    ctx, span := gtrace.NewSpan(gctx.New(), "StartRequests")
    defer span.End()

    ctx = gtrace.SetBaggageValue(ctx, "name", "john")

    content := g.Client().GetContent(ctx, "http://127.0.0.1:8199/hello")
    g.Log().Print(ctx, content)
}
```

1. jaeger.InitJaeger
2. gtrace.SetBaggageValue(ctx, "name", "john")baggagebaggagebaggagecontext.Context
3. g.Client()HTTP
4. g.Log().Print(ctx, content)ctxctxTraceId

Content Menu

- baggage
-
-
-
- HTTP Client Attributes
- HTTP Client Events
- HTTP Server Attributes
- HTTP Server Events

```

package main

import (
    "github.com/gogf/gf/contrib/trace/otlphttp/v2"
    "github.com/gogf/gf/v2/frame/g"
    "github.com/gogf/gf/v2/net/ghttp"
    "github.com/gogf/gf/v2/net/gtrace"
    "github.com/gogf/gf/v2/os/gctx"
)

const (
    serviceName = "otlp-http-server"
    endpoint    = "tracing-analysis-dc-hz.aliyuncs.com"
    path        = "adapt_*****_*****/api/otlp/traces" )

func main() {
    var ctx = gctx.New()
    shutdown, err := otlphttp.Init(serviceName, endpoint, path)
    if err != nil {
        g.Log().Fatal(ctx, err)
    }
    defer shutdown()

    s := g.Server()
    s.Group("/", func(group *ghttp.RouterGroup) {
        group.GET("/hello", HelloHandler)
    })
    s.SetPort(8199)
    s.Run()
}

func HelloHandler(r *ghttp.Request) {
    ctx, span := gtrace.NewSpan(r.Context(), "HelloHandler")
    defer span.End()

    value := gtrace.GetBaggageVar(ctx, "name").String()

    r.Response.Write("hello:", value)
}

```

1. jaeger.InitJaeger
- 2.
3. gtrace.GetBaggageVar(ctx, "name").String()baggage

```
➔ gf-tracing git:(master) go run http/server/main.go
```

SERVER	DOMAIN	ADDRESS	METHOD	ROUTE	HANDLER	MIDDLEWARE
default	default	:8199	GET	/hello	main.HelloHandler	ghttp.MiddlewareServerTracing

```
2021-01-29 00:50:21.649 67782: http server started listening on [:8199]
```

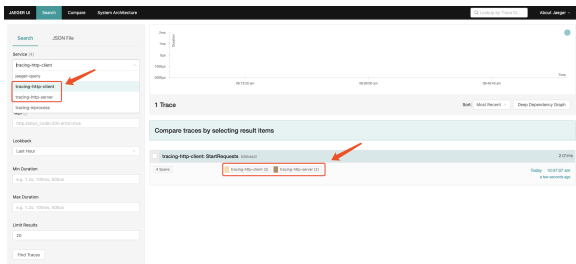
```
➔ gf-tracing git:(master) go run http/client/main.go
```

```
2021-01-29 00:50:34.568 {TraceID:71d3dd5d82d801aea6888f5811dd451b} hello:john
```

```
➔ gf-tracing git:(master)
```

```
baggageTraceIdTraceIdIDIDTraceIdJaeger
```

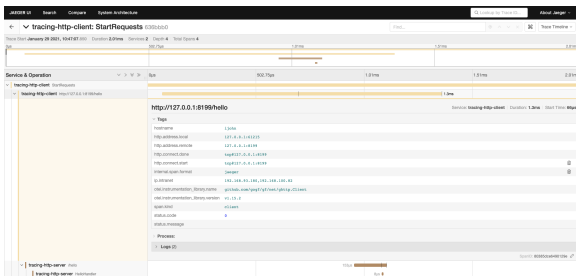
```
Jaeger
```



tracing-http-clienttracing-http-serverHTTP2span

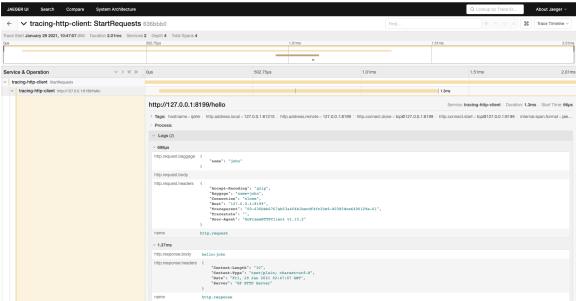
traceAttributesEventsJaegerTagsProcess

HTTP Client Attributes



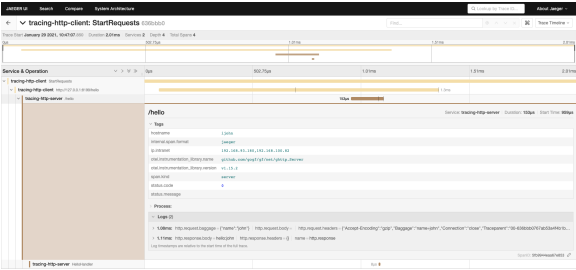
Attribute/Tag															
otel.instrumentation_library.name	span														
otel.instrumentation_library.version															
span.kind	span span <table> <tr> <td></td><td></td></tr> <tr> <td>client</td><td></td></tr> <tr> <td>server</td><td></td></tr> <tr> <td>producer</td><td>MQ</td></tr> <tr> <td>consumer</td><td>MQ</td></tr> <tr> <td>internal</td><td></td></tr> <tr> <td>undefined</td><td></td></tr> </table>			client		server		producer	MQ	consumer	MQ	internal		undefined	
client															
server															
producer	MQ														
consumer	MQ														
internal															
undefined															
status.code	span00														
status.message	span														
hostname															
ip.intranet															
http.address.local	HTTP														
http.address.remote	HTTP														
http.dns.start															
http.dns.done	IP														
http.connect.start															
http.connect.done															

HTTP Client Events



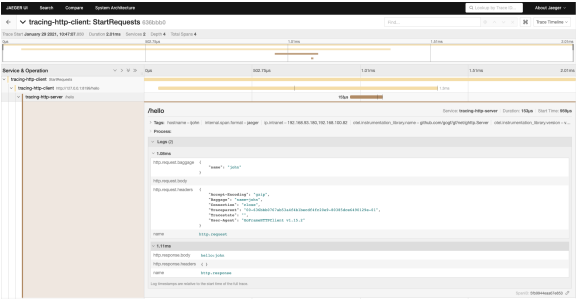
Event/Log	
http.request.headers	HTTPHeader
http.request.baggage	HTTPBaggage
http.request.body	HTTPBody512KB
http.response.headers	HTTPHeader
http.response.body	HTTPBody512KB

HTTP Server Attributes



HTTP ServerAttributesHTTP Client

HTTP Server Events



HTTP ServerEventsHTTP Client