

Tutorial 2: Backend Access

This second tutorial covers the basic of logging the backend calls from within transactions and shows how easy it is to log backend calls of e.g. filesystem, databases, SAP, These logs are stored in the transaction log files and can be analyzed the same way using the [E2E Dashboard](#) services.

Building a Simple Example

Start by using the project from [Tutorial 1: Logging the Transactions of a Simple HTTP Server](#).

Adding a backend call

Instead of writing the response directly, you will read it from a file. Create a file named **Hello.html** and put some HTML data in it. Replace the code of the **sayHello** function to read the response from this file.

```
function sayHello(req, res, cb){
  fs.readFile(__dirname + '/Hello.html',function(err, data){
    res.writeHead(200, {'Content-Type': 'text/html'});
    res.end(data);
    cb();
  });
}
```

Logging the requests

Now use the **e2e-transaction-logger** to trace the backend call. As you need the **trx** object, you have to add a parameter.

```
function sayHello(req, res, trx, cb){
  var io = trx.startIO('Read','FILE','Hello.html');
  fs.readFile(__dirname + '/Hello.html',function(err, data){
    io.end('OK');
    res.writeHead(200, {'Content-Type': 'text/html'});
    res.end(data);
    cb();
  });
}
...
    trx = e2eLogger.startTransaction('Say Hello');
    sayHello(req, res, trx, function(){
      trx.end();
    });
```

Use the **startIO** function of the transaction at the beginning to create the backend call. Parameters of these function are the name of the backend call ('Read'), the domain ('FILE') and the backend system ('Hello.html'). Simply call end on the returned object when it is done.

If you run the server and do some '/hello' requests, you will see the IO logs in the transaction log file.

On this Page:

- [Building a Simple Example](#)
 - [Adding a backend call](#)
 - [Logging the requests](#)
 - [Transaction log files: IO entries](#)
- [Extending the Logging](#)
 - [Multiple IO calls](#)
 - [Success or fail](#)
- [Dashboards](#)

Related Pages:

- [Documentation of the E2E Transaction Logger](#)
- [Tutorial 1: Logging the Transactions of a Simple HTTP Server](#)
- [Tutorial 2: Backend Access](#)
- [Tutorial 3: Express](#)

- [Scheer E2E Dashboards](#)

[Download Example Project](#)

```

2014-04-22    08:52:48    +0200    64a173f0-b9c4-4adc-b598-28592bd4a1d3
1    Say Hello    0    OK    INTERFACE    SERVICE_ENTER
2014-04-22    08:52:48    +0200    64a173f0-b9c4-4adc-b598-28592bd4a1d3
1    Read    2    OK    FILE    IO_ENTER    Hello.html
2014-04-22    08:52:48    +0200    64a173f0-b9c4-4adc-b598-28592bd4a1d3
1    Read    4    OK    FILE    IO_EXIT    Hello.html
2014-04-22    08:52:48    +0200    64a173f0-b9c4-4adc-b598-28592bd4a1d3
1    Say Hello    6    OK    INTERFACE    SERVICE_EXIT
2014-04-22    08:52:52    +0200    eba414a6-bd61-4b1d-b870-81757f74d54b
2    Say Hello    0    OK    INTERFACE    SERVICE_ENTER
2014-04-22    08:52:52    +0200    eba414a6-bd61-4b1d-b870-81757f74d54b
2    Read    1    OK    FILE    IO_ENTER    Hello.html
2014-04-22    08:52:52    +0200    eba414a6-bd61-4b1d-b870-81757f74d54b
2    Read    2    OK    FILE    IO_EXIT    Hello.html
2014-04-22    08:52:52    +0200    eba414a6-bd61-4b1d-b870-81757f74d54b
2    Say Hello    3    OK    INTERFACE    SERVICE_EXIT
2014-04-22    08:52:56    +0200    ee13c258-e2f9-4a78-ba7f-0071a1e5e81e
3    Say Hello    0    OK    INTERFACE    SERVICE_ENTER
2014-04-22    08:52:56    +0200    ee13c258-e2f9-4a78-ba7f-0071a1e5e81e
3    Read    1    OK    FILE    IO_ENTER    Hello.html
2014-04-22    08:52:56    +0200    ee13c258-e2f9-4a78-ba7f-0071a1e5e81e
3    Read    1    OK    FILE    IO_EXIT    Hello.html
2014-04-22    08:52:56    +0200    ee13c258-e2f9-4a78-ba7f-0071a1e5e81e
3    Say Hello    2    OK    INTERFACE    SERVICE_EXIT

```

Transaction log files: IO entries

The parameters are in the following order:

Parameter	Format	Description
Date	YYYY-MM-DD	The current date.
Time	HH:mm:s	The time.
Timezone	+/-HHMM	The timezone offset.
Transaction ID	string	The transaction id is defined by the developer or uniquely generated.
Request ID	number	This is the unique id of the request on this server. It is automatically generated.
Name	string	The name of this IO, defined by the developer.
Elapsed time	number	The number of milliseconds elapsed since the start of the transaction.
State	string	The state of the entry. Can be 'OK' or 'ERROR'. It is used to differentiate successful operations and failed ones.
Domain	string	The domain indicates which kind of entry it is. In case of IOs, it is defined by the developer.
Log type	string	The log type indicates the type of entry. In case of IOs, it can be 'IO_ENTER' on IO call start and 'IO_EXIT' on IO call end.
Parameter 1		Parameter 1 is not used by IO entries.
Parameter 2	string	For the IO calls, parameter 2 contains the backend system as provided by the developer.

Extending the Logging

Multiple IO calls

Let's do some more calls in the **doSomething** request as it could be done in a more realistic server.

```
function doSomething(req, res, trx, cb){
  var io = trx.startIO('GetCustomer', 'SAP', 'http://sap.e2e.ch:3000/');
  sapGetCustomer(1, function(err, customer){
    io.end();

    io = trx.startIO('GetCustomerDetail', 'DB', 'localhost/mongodb/');
    dbGetCustomerDetail(customer, function(err, customer){
      io.end();

      res.writeHead(200, {'Content-Type': 'application/json'});
      res.end(JSON.stringify(customer));

      cb();
    });
  });
}
```

And you will get some logs looking like this.

```

2014-04-22 09:29:59 +0200 1b50ca29-febc-4672-a6b9-faac1eceba83
2 Say Hello 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:29:59 +0200 1b50ca29-febc-4672-a6b9-faac1eceba83
2 Read 0 OK FILE IO_ENTER Hello.html
2014-04-22 09:29:59 +0200 1b50ca29-febc-4672-a6b9-faac1eceba83
2 Read 1 OK FILE IO_EXIT Hello.html
2014-04-22 09:29:59 +0200 1b50ca29-febc-4672-a6b9-faac1eceba83
2 Say Hello 2 OK INTERFACE SERVICE_EXIT
2014-04-22 09:30:07 +0200 b88f795f-3141-4cf0-9cdc-2dfbd7d612b2
3 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:30:07 +0200 b88f795f-3141-4cf0-9cdc-2dfbd7d612b2
3 GetCustomer 0 OK SAP IO_ENTER http://sap.e2e.ch:
3000/
2014-04-22 09:30:07 +0200 b88f795f-3141-4cf0-9cdc-2dfbd7d612b2
3 GetCustomer 239 OK SAP IO_EXIT http://sap.e2e.ch:
3000/
2014-04-22 09:30:07 +0200 b88f795f-3141-4cf0-9cdc-2dfbd7d612b2
3 GetCustomerDetail 239 OK DB IO_ENTER localhost
/mongodb/
2014-04-22 09:30:08 +0200 b88f795f-3141-4cf0-9cdc-2dfbd7d612b2
3 GetCustomerDetail 1014 OK DB IO_EXIT localhost
/mongodb/
2014-04-22 09:30:08 +0200 b88f795f-3141-4cf0-9cdc-2dfbd7d612b2
3 Do Something 1015 OK INTERFACE SERVICE_EXIT
2014-04-22 09:30:13 +0200 79686607-b17e-4a78-982c-22479dc97394
4 Say Hello 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:30:13 +0200 79686607-b17e-4a78-982c-22479dc97394
4 Read 0 OK FILE IO_ENTER Hello.html
2014-04-22 09:30:13 +0200 79686607-b17e-4a78-982c-22479dc97394
4 Read 1 OK FILE IO_EXIT Hello.html
2014-04-22 09:30:13 +0200 79686607-b17e-4a78-982c-22479dc97394
4 Say Hello 2 OK INTERFACE SERVICE_EXIT
2014-04-22 09:30:17 +0200 f7e554f8-75e1-4bf7-9a9e-f738872d7f84
5 Say Hello 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:30:17 +0200 f7e554f8-75e1-4bf7-9a9e-f738872d7f84
5 Read 1 OK FILE IO_ENTER Hello.html
2014-04-22 09:30:17 +0200 f7e554f8-75e1-4bf7-9a9e-f738872d7f84
5 Read 2 OK FILE IO_EXIT Hello.html
2014-04-22 09:30:17 +0200 f7e554f8-75e1-4bf7-9a9e-f738872d7f84
5 Say Hello 3 OK INTERFACE SERVICE_EXIT
2014-04-22 09:30:24 +0200 82279774-459a-4326-9045-a1618047f369
6 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:30:24 +0200 82279774-459a-4326-9045-a1618047f369
6 GetCustomer 0 OK SAP IO_ENTER http://sap.e2e.ch:
3000/
2014-04-22 09:30:26 +0200 82279774-459a-4326-9045-a1618047f369
6 GetCustomer 2024 OK SAP IO_EXIT http://sap.e2e.ch:
3000/
2014-04-22 09:30:26 +0200 82279774-459a-4326-9045-a1618047f369
6 GetCustomerDetail 2024 OK DB IO_ENTER localhost
/mongodb/
2014-04-22 09:30:26 +0200 82279774-459a-4326-9045-a1618047f369
6 GetCustomerDetail 2218 OK DB IO_EXIT localhost
/mongodb/
2014-04-22 09:30:26 +0200 82279774-459a-4326-9045-a1618047f369
6 Do Something 2219 OK INTERFACE SERVICE_EXIT
2014-04-22 09:30:41 +0200 223717d1-0579-4b75-a372-8e85c120fa23
7 Say Good Bye 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:30:41 +0200 223717d1-0579-4b75-a372-8e85c120fa23
7 Say Good Bye 1 OK INTERFACE SERVICE_EXIT

```

Success or fail

The end function of a IO takes a parameter which allow you to set the success or fail of the transaction. The valid values are true or 'OK' for success and false or 'ERROR' for fail. A failed backend call doesn't always mean a failed transaction - this is up to the developer.

```

function doSomething(req, res, trx, cb){Tutorial2.zip
  var io = trx.startIO('GetCustomer', 'SAP', 'http://sap.e2e.ch:3000/');
  sapGetCustomer(1, function(err, customer){
    if(err){
      io.end(false);
      res.writeHead(500, {'Content-Type': 'application/json'});
      res.end(JSON.stringify(err));

      cb(false);
      return;
    }
    io.end();

    io = trx.startIO('GetCustomerDetail', 'DB', 'localhost/mongodb/');
    dbGetCustomerDetail(customer, function(err, customerWithDetail){
      if(err){
        io.end(false);
        res.writeHead(200, {'Content-Type': 'application/json'});
        res.end(JSON.stringify(customer));

        cb();
        return;
      }
      io.end();

      res.writeHead(200, {'Content-Type': 'application/json'});
      res.end(JSON.stringify(customerWithDetail));

      cb();

    });
  });
}

```

Now, you can see the state of the end entries.

```

2014-04-22 09:47:30 +0200 23662f11-1012-40cd-97ed-883ee72f9bb5
5 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:47:30 +0200 23662f11-1012-40cd-97ed-883ee72f9bb5
5 GetCustomer 0 OK SAP IO_ENTER http://sap.e2e.ch:
3000/
2014-04-22 09:47:32 +0200 23662f11-1012-40cd-97ed-883ee72f9bb5
5 GetCustomer 1169 OK SAP IO_EXIT http://sap.e2e.ch:
3000/
2014-04-22 09:47:32 +0200 23662f11-1012-40cd-97ed-883ee72f9bb5
5 GetCustomerDetail 1169 OK DB IO_ENTER localhost
/mongodb/
2014-04-22 09:47:33 +0200 23662f11-1012-40cd-97ed-883ee72f9bb5
5 GetCustomerDetail 2057 OK DB IO_EXIT localhost
/mongodb/
2014-04-22 09:47:33 +0200 23662f11-1012-40cd-97ed-883ee72f9bb5
5 Do Something 2058 OK INTERFACE SERVICE_EXIT
2014-04-22 09:47:39 +0200 9098b116-0fe1-49d5-94fe-994acb7630e0
6 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:47:39 +0200 9098b116-0fe1-49d5-94fe-994acb7630e0
6 GetCustomer 0 OK SAP IO_ENTER http://sap.e2e.ch:
3000/
2014-04-22 09:47:39 +0200 9098b116-0fe1-49d5-94fe-994acb7630e0
6 GetCustomer 206 ERROR SAP IO_EXIT http://sap.e2e.
ch:3000/
2014-04-22 09:47:39 +0200 9098b116-0fe1-49d5-94fe-994acb7630e0
6 Do Something 207 ERROR INTERFACE SERVICE_EXIT
2014-04-22 09:47:44 +0200 b86575fc-025a-4ce8-afbc-7257eddf4fc1
7 Do Something 0 OK INTERFACE SERVICE_ENTER
2014-04-22 09:47:44 +0200 b86575fc-025a-4ce8-afbc-7257eddf4fc1
7 GetCustomer 0 OK SAP IO_ENTER http://sap.e2e.ch:
3000/
2014-04-22 09:47:46 +0200 b86575fc-025a-4ce8-afbc-7257eddf4fc1
7 GetCustomer 2314 OK SAP IO_EXIT http://sap.e2e.ch:
3000/
2014-04-22 09:47:46 +0200 b86575fc-025a-4ce8-afbc-7257eddf4fc1
7 GetCustomerDetail 2314 OK DB IO_ENTER localhost
/mongodb/
2014-04-22 09:47:47 +0200 b86575fc-025a-4ce8-afbc-7257eddf4fc1
7 GetCustomerDetail 3268 ERROR DB IO_EXIT localhost
/mongodb/
2014-04-22 09:47:47 +0200 b86575fc-025a-4ce8-afbc-7257eddf4fc1
7 Do Something 3269 OK INTERFACE SERVICE_EXIT

```

You can download the complete [example project of tutorial 2](#).

Dashboards

These log files are fully compatible with the [E2E Dashboards](#) services, so they can be loaded into the database and used with the dashboards to analyze the performance of your services.

Find below some examples of the views you would get with this enhanced HTTP server.

Figure: Table View of the E2E Service Dashboard

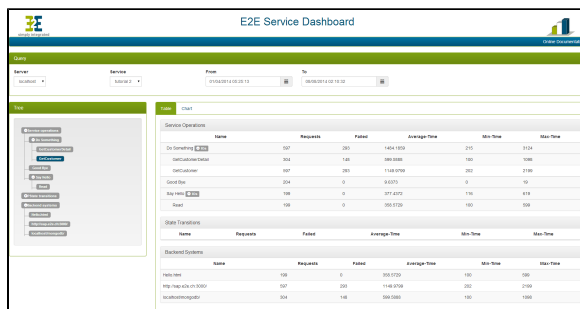


Figure: Graphical View of the E2E Service Dashboard

