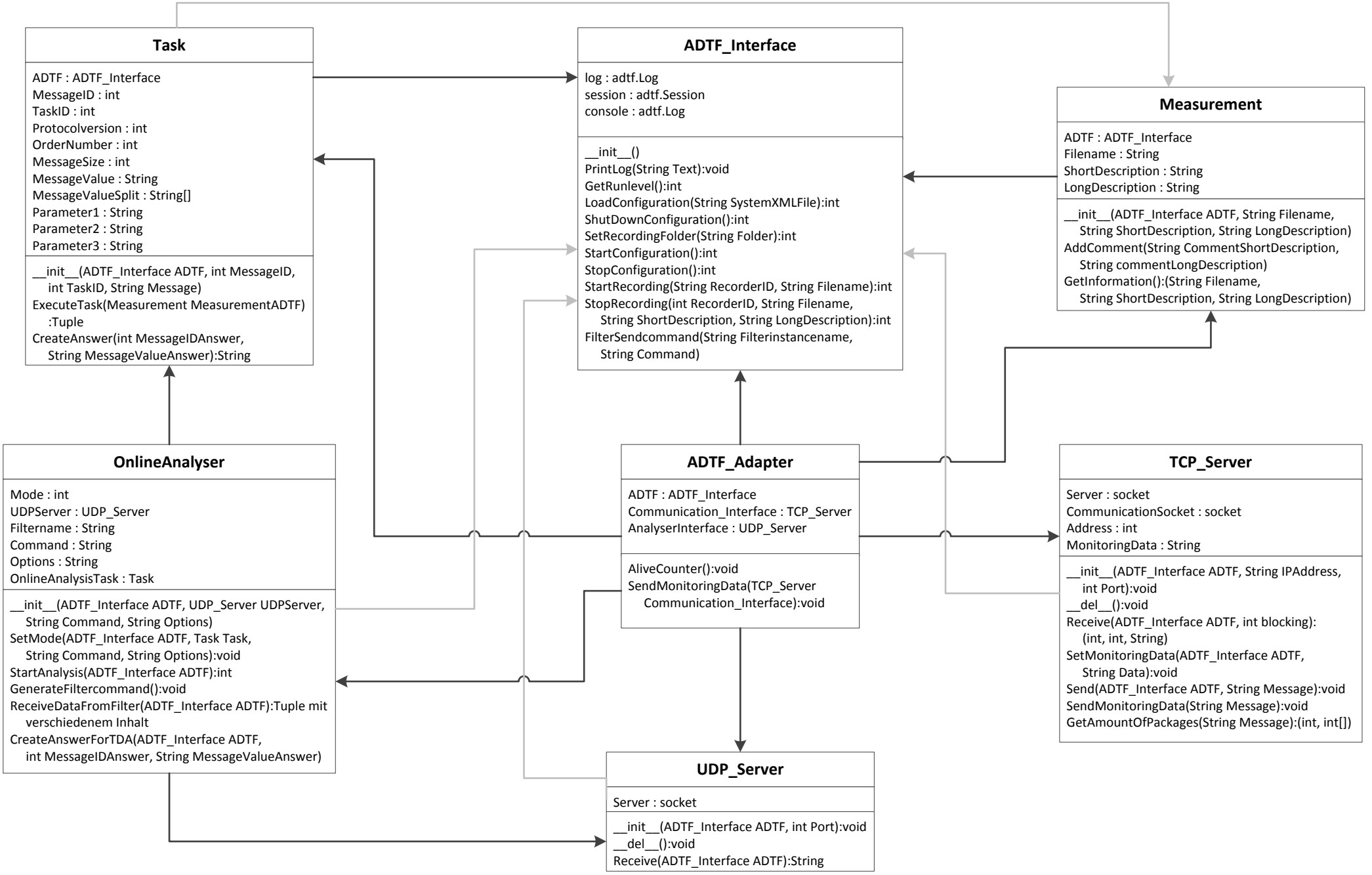
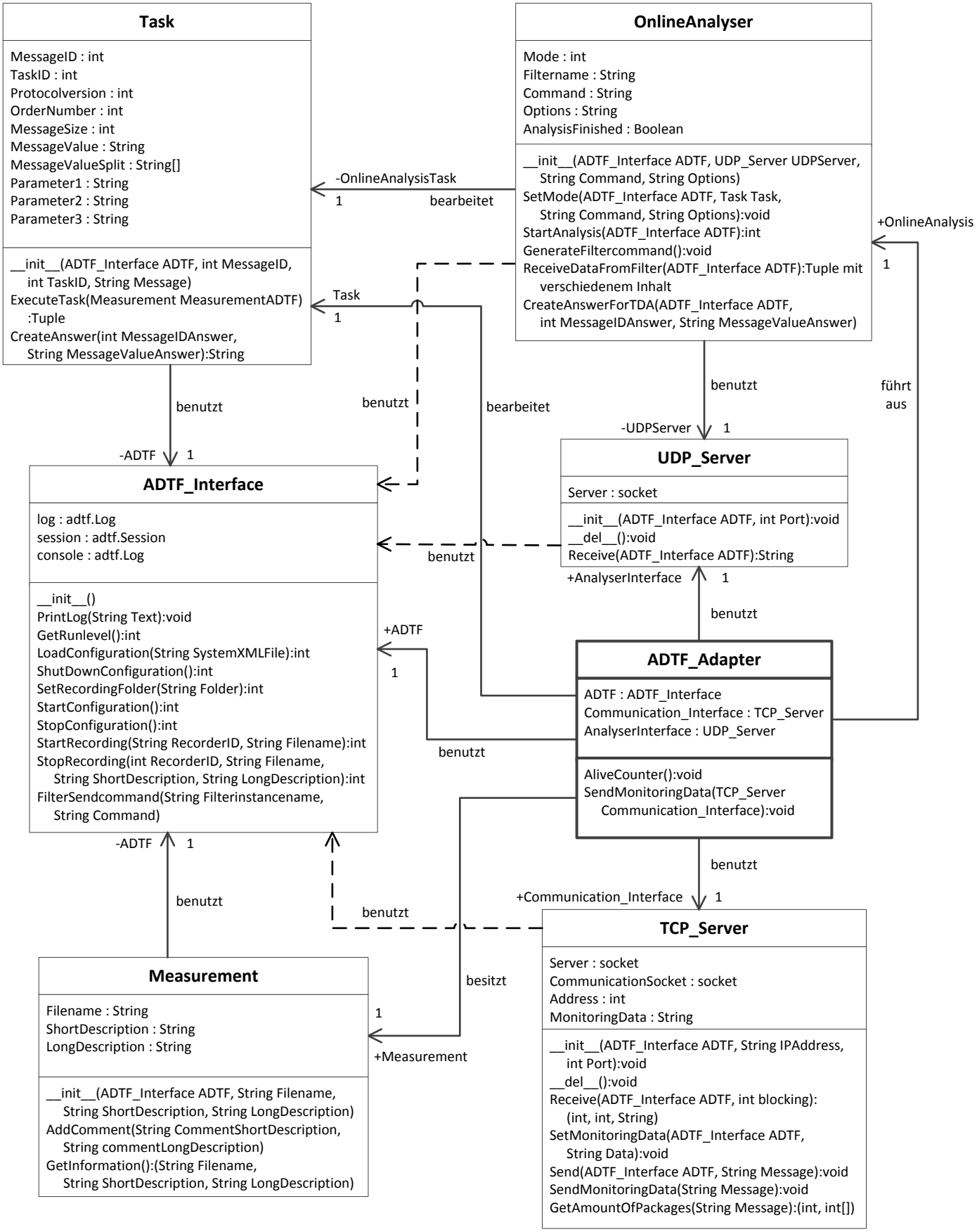
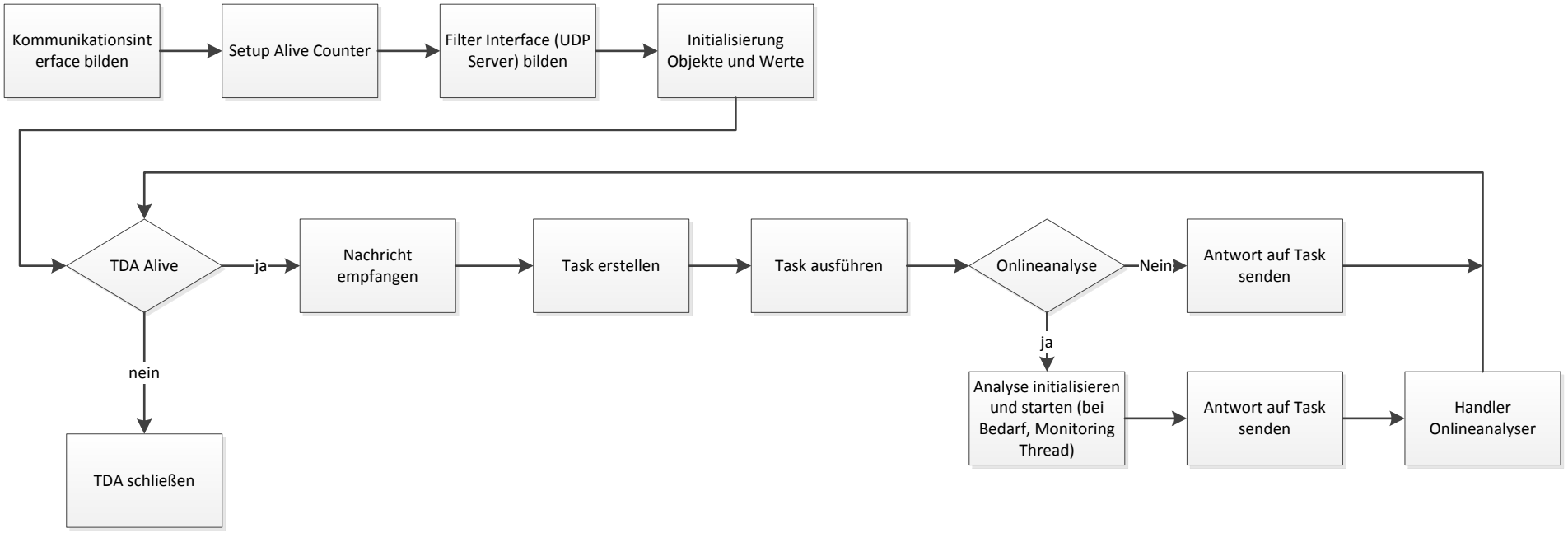
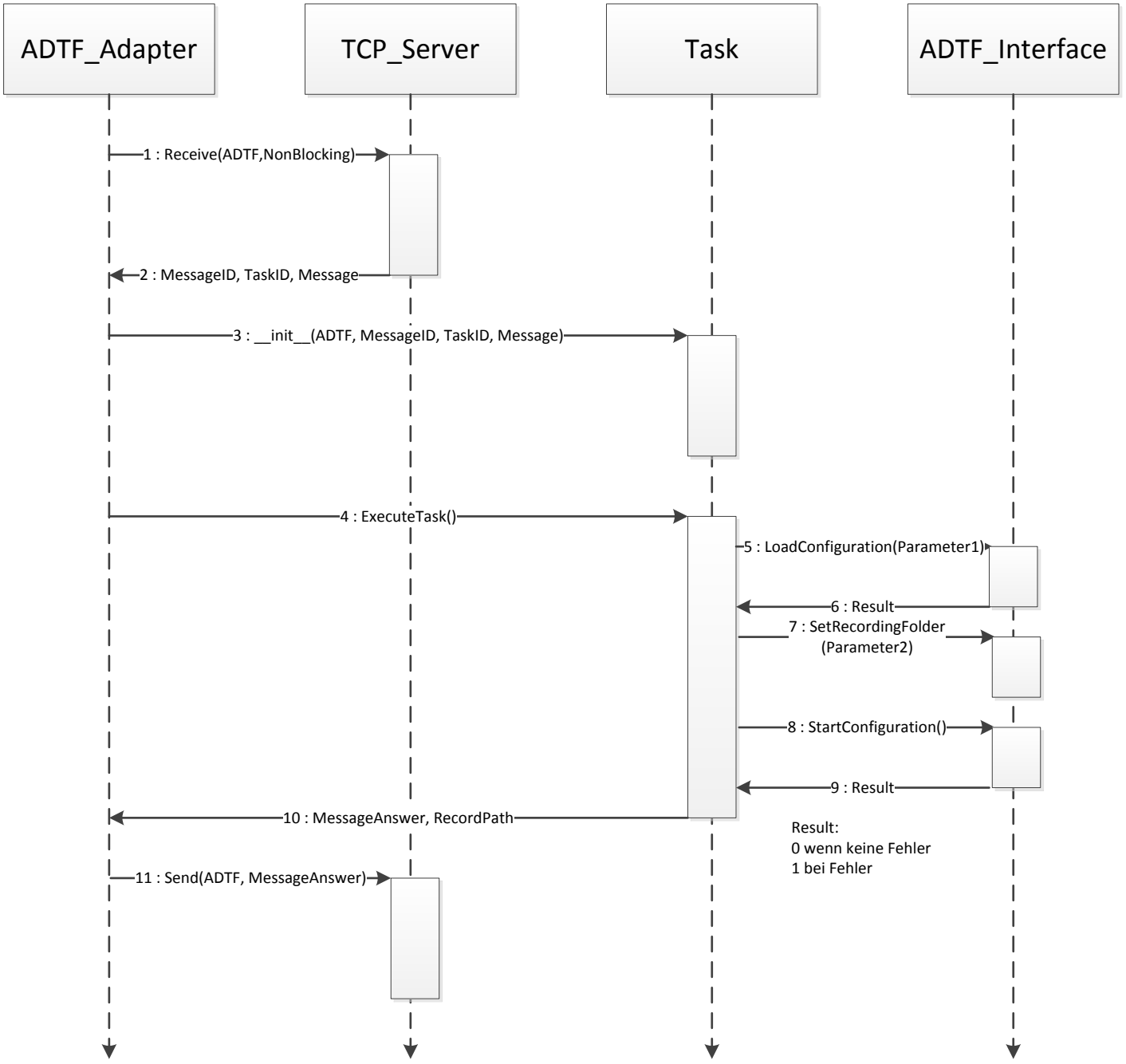
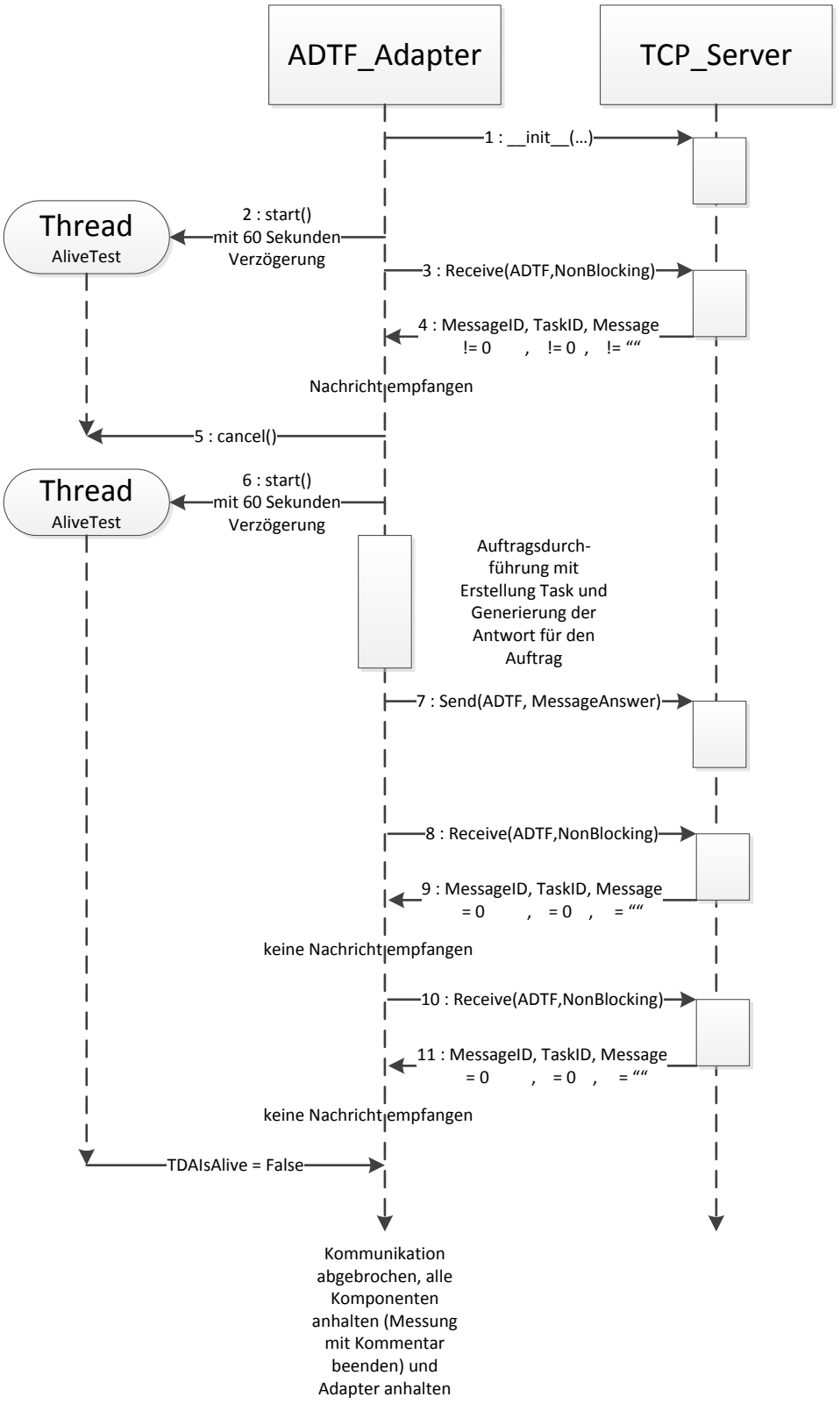

Task
ADTF : ADTF_Interface
MessageID : int
TaskID : int
Protocolversion : int
OrderNumber : int
MessageSize : int
MessageValue : String
MessageValueSplit : String[]
Parameter1 : String
Parameter2 : String
Parameter3 : String
__init__(ADTF_Interface ADTF, int MessageID, int TaskID, String Message)
ExecuteTask(Measurement MeasurementADTF) :Tuple
CreateAnswer(int MessageIDAnswer, String MessageValueAnswer):String
ADTF_Interface
log : adtf.Log
session : adtf.Session
console : adtf.Log
__init__()
PrintLog(String Text):void
GetRunlevel():int
LoadConfiguration(String SystemXMLFile):int
ShutDownConfiguration():int
SetRecordingFolder(String Folder):int
StartConfiguration():int
StopConfiguration():int
StartRecording(String RecorderID, String Filename):int
StopRecording(int RecorderID, String Filename, String ShortDescription, String LongDescription):int
FilterSendcommand(String Filterinstancename, String Command)
Measurement
ADTF : ADTF_Interface
Filename : String
ShortDescription : String
LongDescription : String
__init__(ADTF_Interface ADTF, String Filename, String ShortDescription, String LongDescription)
AddComment(String CommentShortDescription, String commentLongDescription)
GetInformation():(String Filename, String ShortDescription, String LongDescription)
OnlineAnalyser
Mode : int
UDPServer : UDP_Server
Filtername : String
Command : String
Options : String
OnlineAnalysisTask : Task
__init__(ADTF_Interface ADTF, UDP_Server UDPServer, String Command, String Options)
SetMode(ADTF_Interface ADTF, Task Task, String Command, String Options):void
StartAnalysis(ADTF_Interface ADTF):int
GenerateFiltercommand():void
ReceiveDataFromFilter(ADTF_Interface ADTF):Tuple mit verschiedenem Inhalt
CreateAnswerForTDA(ADTF_Interface ADTF, int MessageIDAnswer, String MessageValueAnswer)
ADTF_Adapter
ADTF : ADTF_Interface
Communication_Interface : TCP_Server
AnalyserInterface : UDP_Server
AliveCounter():void
SendMonitoringData(TCP_Server Communication_Interface):void
TCP_Server
Server : socket
CommunicationSocket : socket
Address : int
MonitoringData : String
__init__(ADTF_Interface ADTF, String IPAddress, int Port):void
__del__():void
Receive(ADTF_Interface ADTF, int blocking): (int, int, String)
SetMonitoringData(ADTF_Interface ADTF, String Data):void
Send(ADTF_Interface ADTF, String Message):void
SendMonitoringData(String Message):void
GetAmountOfPackages(String Message):(int, int[])
UDP_Server
Server : socket
__init__(ADTF_Interface ADTF, int Port):void
__del__():void
Receive(ADTF_Interface ADTF):String

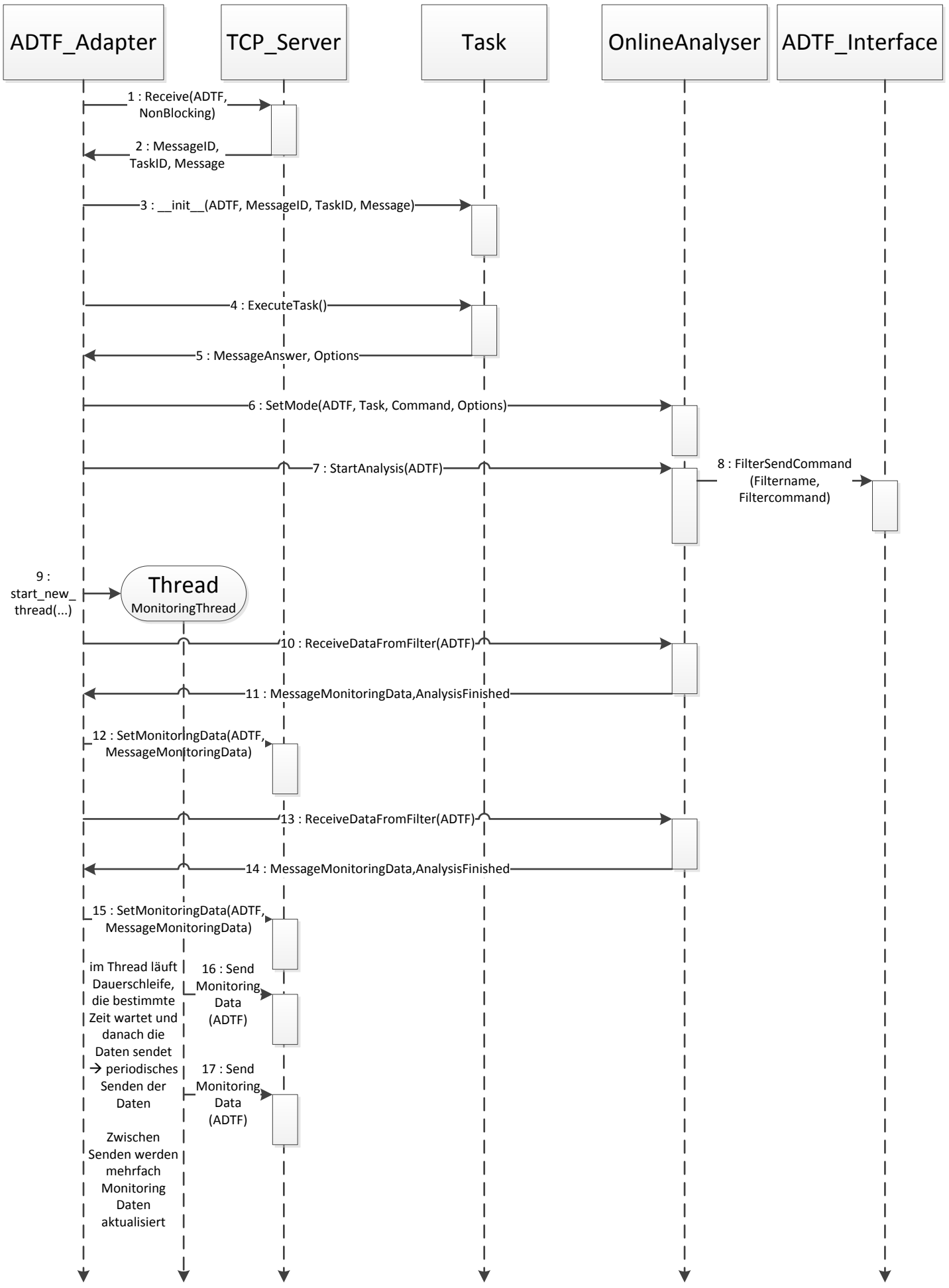

Task
MessageID : int
TaskID : int
Protocolversion : int
OrderNumber : int
MessageSize : int
MessageValue : String
MessageValueSplit : String[]
Parameter1 : String
Parameter2 : String
Parameter3 : String
__init__(ADTF_Interface ADTF, int MessageID, int TaskID, String Message)
ExecuteTask(Measurement MeasurementADTF) :Tuple
CreateAnswer(int MessageIDAnswer, String MessageValueAnswer):String
OnlineAnalyser
Mode : int
Filtername : String
Command : String
Options : String
AnalysisFinished : Boolean
__init__(ADTF_Interface ADTF, UDP_Server UDPServer, String Command, String Options)
SetMode(ADTF_Interface ADTF, Task Task, String Command, String Options):void
StartAnalysis(ADTF_Interface ADTF):int
GenerateFiltercommand():void
ReceiveDataFromFilter(ADTF_Interface ADTF):Tuple mit verschiedenem Inhalt
CreateAnswerForTDA(ADTF_Interface ADTF, int MessageIDAnswer, String MessageValueAnswer)
-OnlineAnalysisTask
1
bearbeitet
+OnlineAnalysis
1
führt aus
Task
1
benutzt
benutzt
bearbeitet
-UDPServer
1
-ADTF
1
ADTF_Interface
log : adtf.Log
session : adtf.Session
console : adtf.Log
__init__()
PrintLog(String Text):void
GetRunlevel():int
LoadConfiguration(String SystemXMLFile):int
ShutDownConfiguration():int
SetRecordingFolder(String Folder):int
StartConfiguration():int
StopConfiguration():int
StartRecording(String RecorderID, String Filename):int
StopRecording(int RecorderID, String Filename, String ShortDescription, String LongDescription):int
FilterSendcommand(String Filterinstancename, String Command)
UDP_Server
Server : socket
__init__(ADTF_Interface ADTF, int Port):void
__del__():void
Receive(ADTF_Interface ADTF):String
benutzt
+AnalyserInterface
1
benutzt
+ADTF
1
ADTF_Adapter
ADTF : ADTF_Interface
Communication_Interface : TCP_Server
AnalyserInterface : UDP_Server
AliveCounter():void
SendMonitoringData(TCP_Server Communication_Interface):void
benutzt
-ADTF
1
benutzt
benutzt
+Communication_Interface
1
Measurement
Filename : String
ShortDescription : String
LongDescription : String
__init__(ADTF_Interface ADTF, String Filename, String ShortDescription, String LongDescription)
AddComment(String CommentShortDescription, String commentLongDescription)
GetInformation():(String Filename, String ShortDescription, String LongDescription)
besitzt
1
+Measurement
TCP_Server
Server : socket
CommunicationSocket : socket
Address : int
MonitoringData : String
__init__(ADTF_Interface ADTF, String IPAddress, int Port):void
__del__():void
Receive(ADTF_Interface ADTF, int blocking): (int, int, String)
SetMonitoringData(ADTF_Interface ADTF, String Data):void
Send(ADTF_Interface ADTF, String Message):void
SendMonitoringData(String Message):void
GetAmountOfPackages(String Message):(int, int[])

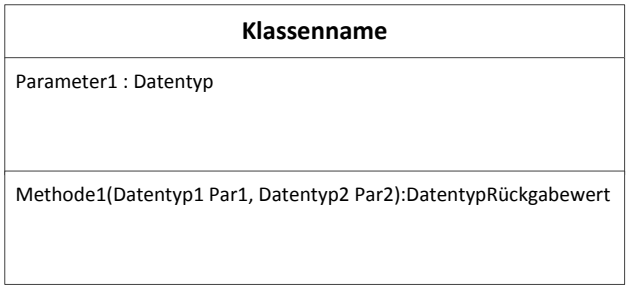

Kommunikationsinterface bilden
Setup Alive Counter
Filter Interface (UDP Server) bilden
Initialisierung Objekte und Werte
TDA Alive
ja
Nachricht empfangen
Task erstellen
Task ausführen
Onlineanalyse
Nein
Antwort auf Task senden
ja
nein
Analyse initialisieren und starten (bei Bedarf, Monitoring Thread)
Antwort auf Task senden
Handler Onlineanalyser
TDA schließen

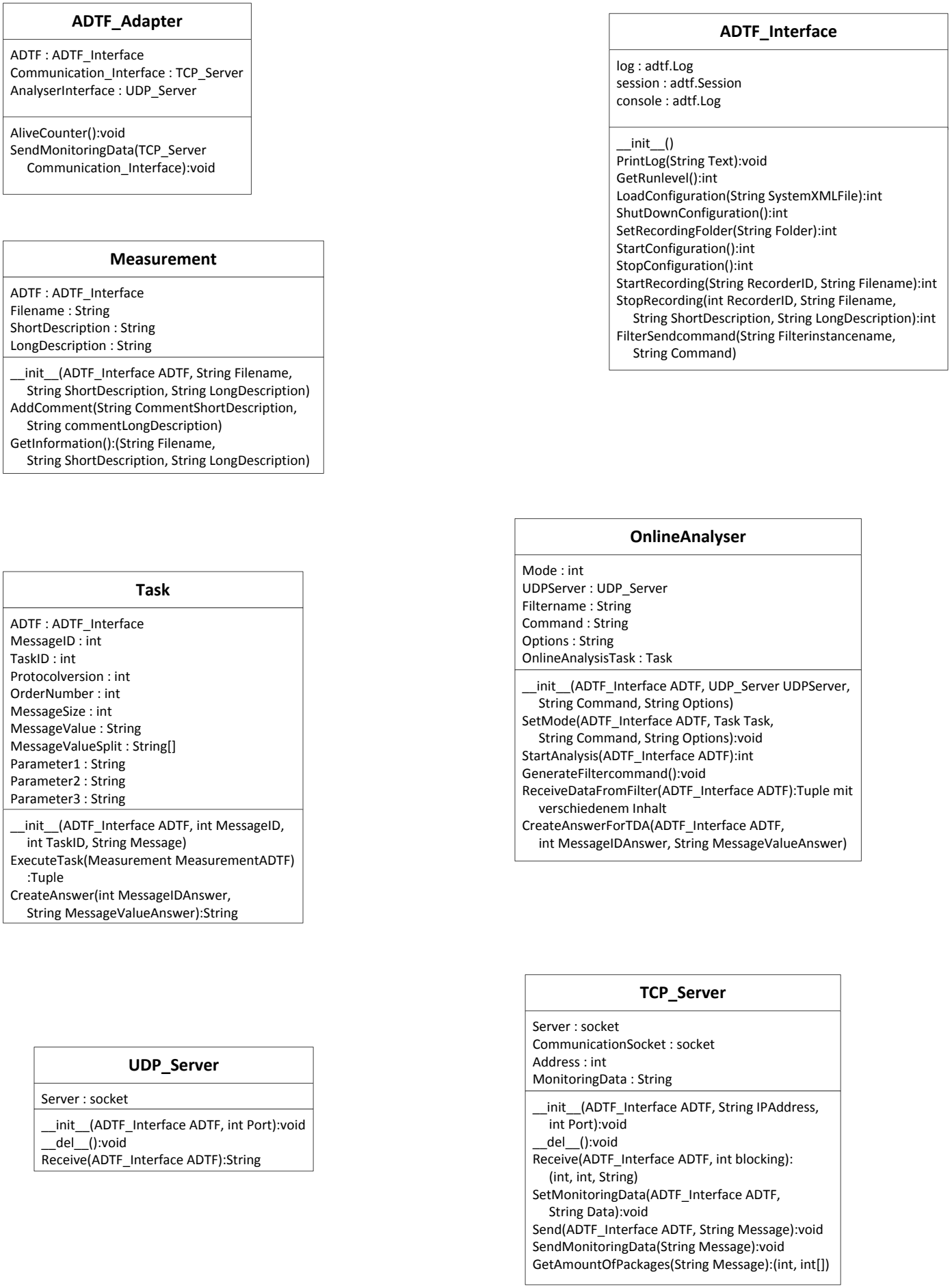
Sequenzdiagramm Load Config: Message ID = 2, Task ID = 1




ADTF_Adapter
TCP_Server
1 : __init__(...)
Thread
AliveTest
2 : start()
mit 60 Sekunden
Verzögerung
3 : Receive(ADTF,NonBlocking)
4 : MessageID, TaskID, Message
!= 0 , != 0 , != ""
Nachricht empfangen
5 : cancel()
Thread
AliveTest
6 : start()
mit 60 Sekunden
Verzögerung
Auftragsdurch-
führung mit
Erstellung Task und
Generierung der
Antwort für den
Auftrag
7 : Send(ADTF, MessageAnswer)
8 : Receive(ADTF,NonBlocking)
9 : MessageID, TaskID, Message
= 0 , = 0 , = ""
keine Nachricht empfangen
10 : Receive(ADTF,NonBlocking)
11 : MessageID, TaskID, Message
= 0 , = 0 , = ""
keine Nachricht empfangen
TDAIsAlive = False
Kommunikation
abgebrochen, alle
Komponenten
anhalten (Messung
mit Kommentar
beenden) und
Adapter anhalten


ADTF_Adapter
TCP_Server
Task
OnlineAnalyser
ADTF_Interface
1 : Receive(ADTF, NonBlocking)
2 : MessageID, TaskID, Message
3 : __init__(ADTF, MessageID, TaskID, Message)
4 : ExecuteTask()
5 : MessageAnswer, Options
6 : SetMode(ADTF, Task, Command, Options)
7 : StartAnalysis(ADTF)
8 : FilterSendCommand (Filtername, Filtercommand)
9 : start_new_ thread(...)
Thread
MonitoringThread
10 : ReceiveDataFromFilter(ADTF)
11 : MessageMonitoringData,AnalysisFinished
12 : SetMonitoringData(ADTF, MessageMonitoringData)
13 : ReceiveDataFromFilter(ADTF)
14 : MessageMonitoringData,AnalysisFinished
15 : SetMonitoringData(ADTF, MessageMonitoringData)
im Thread läuft Dauerschleife, die bestimmte Zeit wartet und danach die Daten sendet → periodisches Senden der Daten
16 : Send Monitoring Data (ADTF)
17 : Send Monitoring Data (ADTF)
Zwischen Senden werden mehrfach Monitoring Daten aktualisiert


Klassenname
Parameter1 : Datentyp
Methode1(Datentyp1 Par1, Datentyp2 Par2):DatentypRückgabewert


ADTF_Adapter
ADTF : ADTF_Interface
Communication_Interface : TCP_Server
AnalyserInterface : UDP_Server
AliveCounter():void
SendMonitoringData(TCP_Server Communication_Interface):void
ADTF_Interface
log : adtf.Log
session : adtf.Session
console : adtf.Log
__init__()
PrintLog(String Text):void
GetRunlevel():int
LoadConfiguration(String SystemXMLFile):int
ShutDownConfiguration():int
SetRecordingFolder(String Folder):int
StartConfiguration():int
StopConfiguration():int
StartRecording(String RecorderID, String Filename):int
StopRecording(int RecorderID, String Filename, String ShortDescription, String LongDescription):int
FilterSendcommand(String Filterinstancename, String Command)
Measurement
ADTF : ADTF_Interface
Filename : String
ShortDescription : String
LongDescription : String
__init__(ADTF_Interface ADTF, String Filename, String ShortDescription, String LongDescription)
AddComment(String CommentShortDescription, String commentLongDescription)
GetInformation():(String Filename, String ShortDescription, String LongDescription)
OnlineAnalyser
Mode : int
UDPServer : UDP_Server
Filtername : String
Command : String
Options : String
OnlineAnalysisTask : Task
__init__(ADTF_Interface ADTF, UDP_Server UDPServer, String Command, String Options)
SetMode(ADTF_Interface ADTF, Task Task, String Command, String Options):void
StartAnalysis(ADTF_Interface ADTF):int
GenerateFiltercommand():void
ReceiveDataFromFilter(ADTF_Interface ADTF):Tuple mit verschiedenem Inhalt
CreateAnswerForTDA(ADTF_Interface ADTF, int MessageIDAnswer, String MessageValueAnswer)
Task
ADTF : ADTF_Interface
MessageID : int
TaskID : int
Protocolversion : int
OrderNumber : int
MessageSize : int
MessageValue : String
MessageValueSplit : String[]
Parameter1 : String
Parameter2 : String
Parameter3 : String
__init__(ADTF_Interface ADTF, int MessageID, int TaskID, String Message)
ExecuteTask(Measurement MeasurementADTF) :Tuple
CreateAnswer(int MessageIDAnswer, String MessageValueAnswer):String
TCP_Server
Server : socket
CommunicationSocket : socket
Address : int
MonitoringData : String
__init__(ADTF_Interface ADTF, String IPAddress, int Port):void
__del__():void
Receive(ADTF_Interface ADTF, int blocking): (int, int, String)
SetMonitoringData(ADTF_Interface ADTF, String Data):void
Send(ADTF_Interface ADTF, String Message):void
SendMonitoringData(String Message):void
GetAmountOfPackages(String Message):(int, int[])
UDP_Server
Server : socket
__init__(ADTF_Interface ADTF, int Port):void
__del__():void
Receive(ADTF_Interface ADTF):String