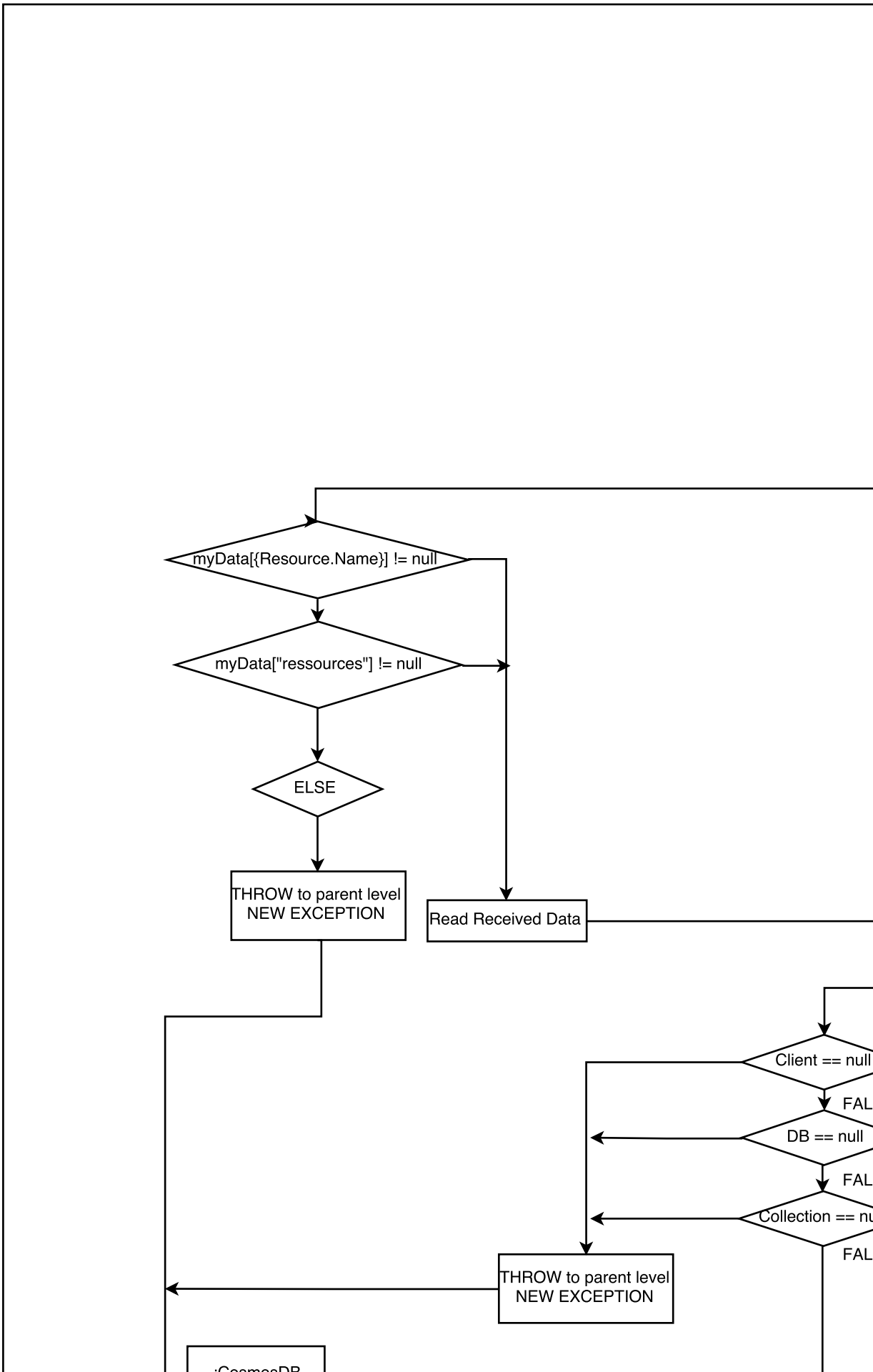
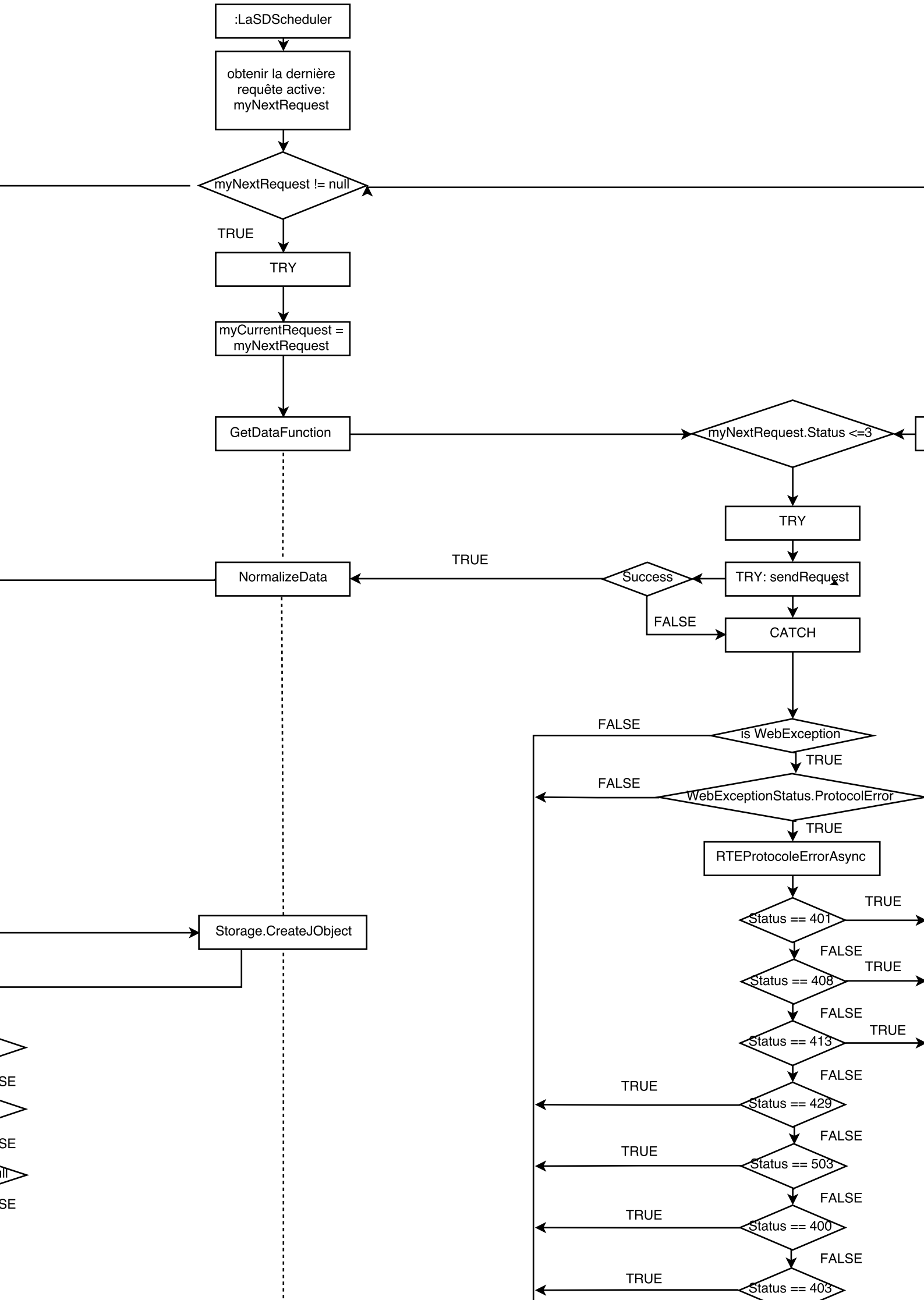
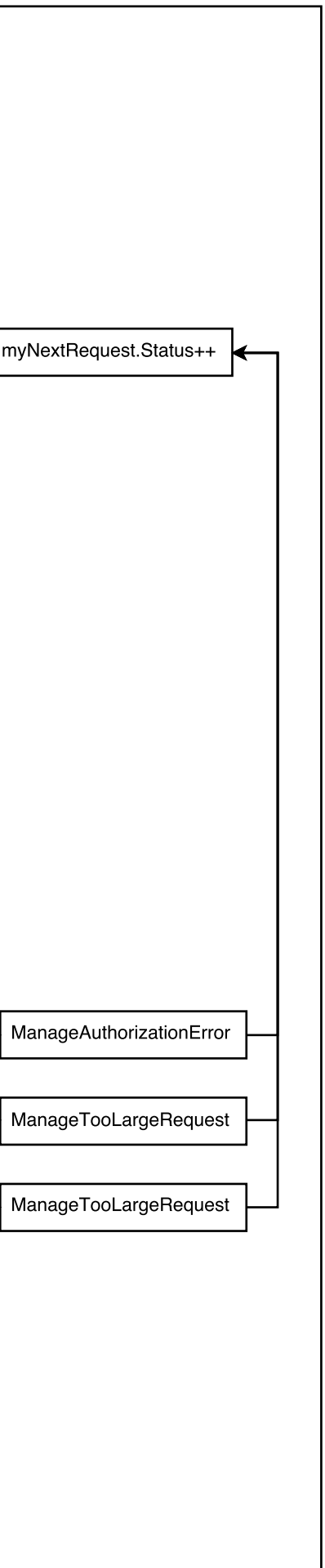
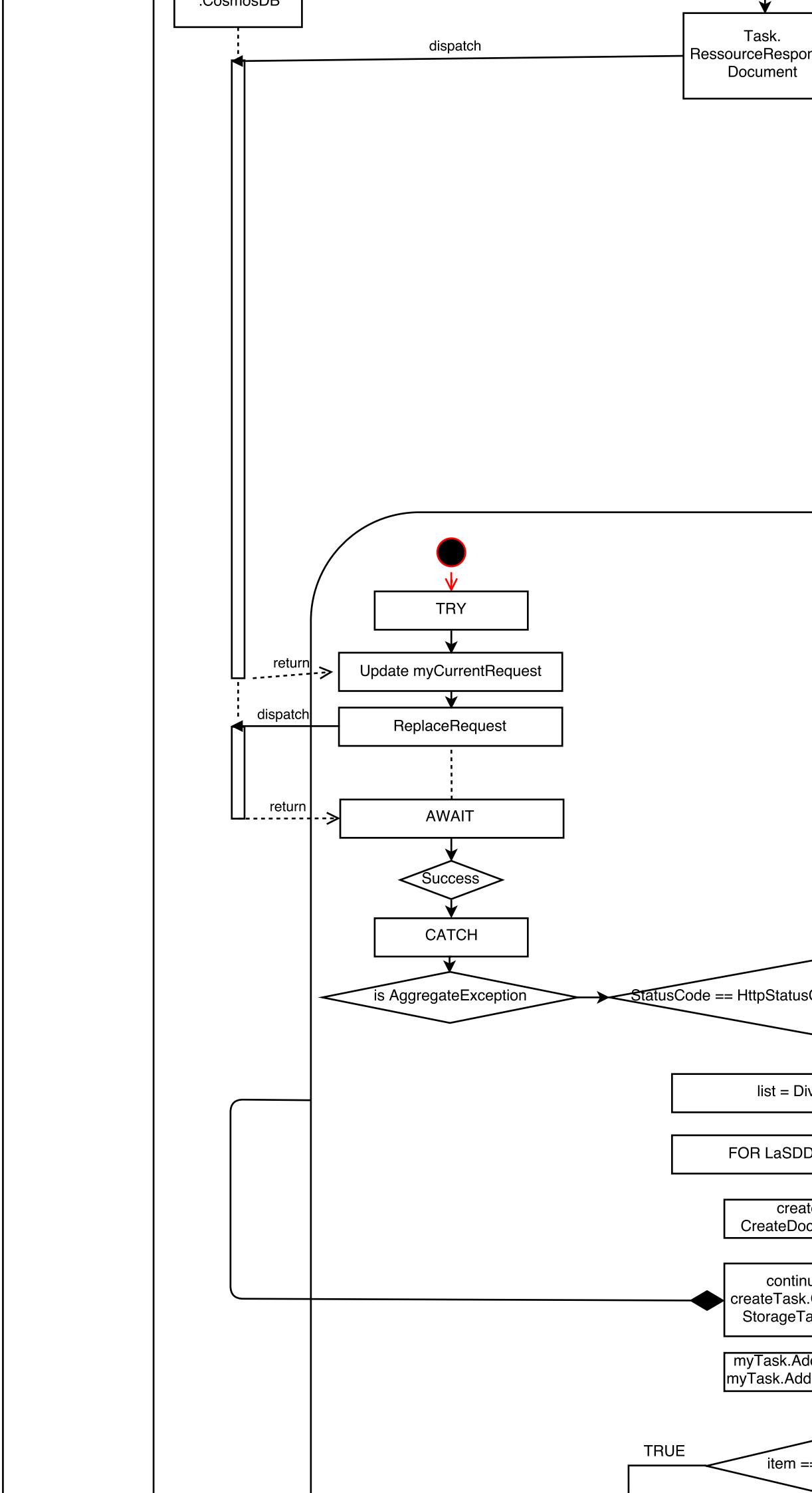
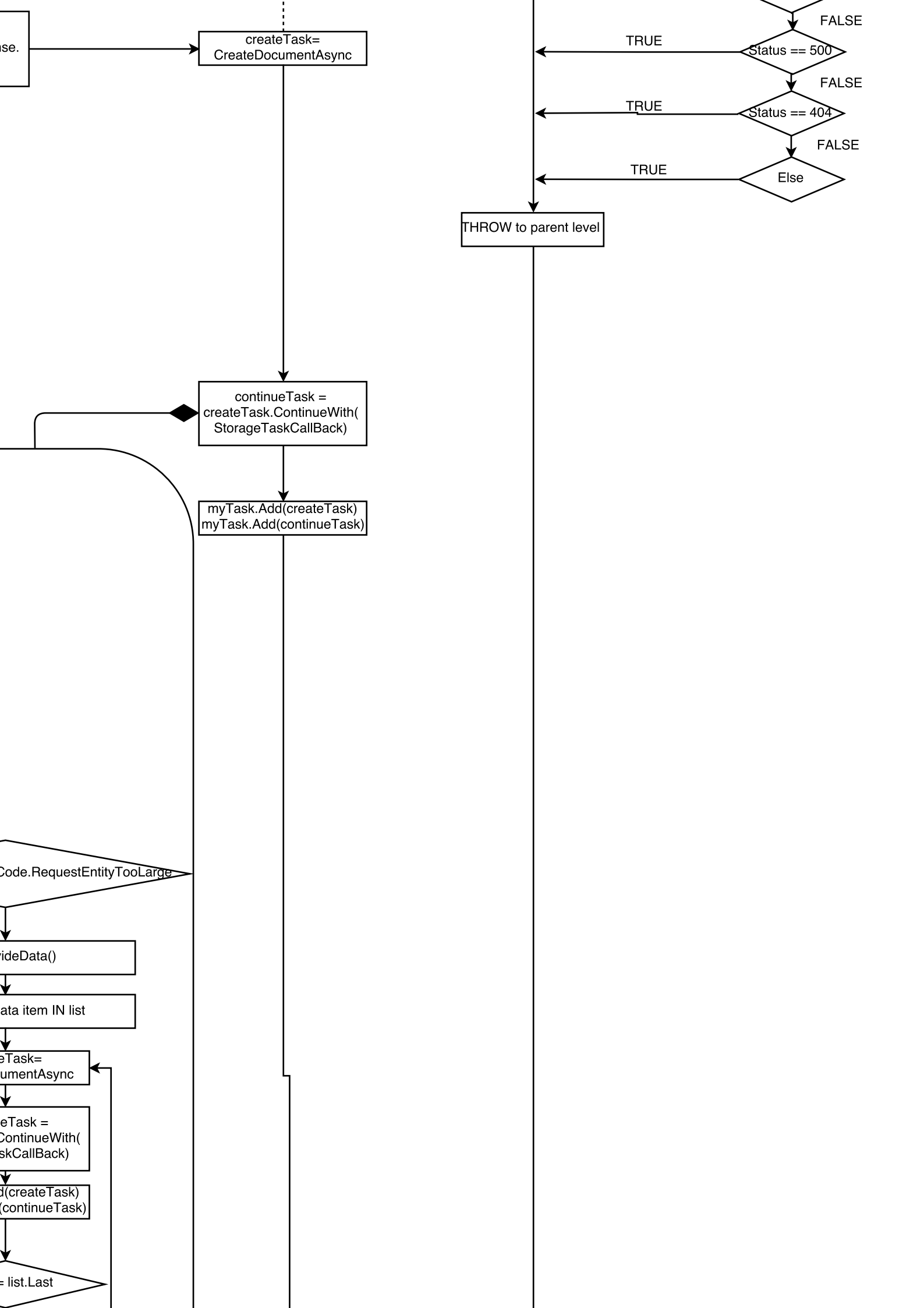
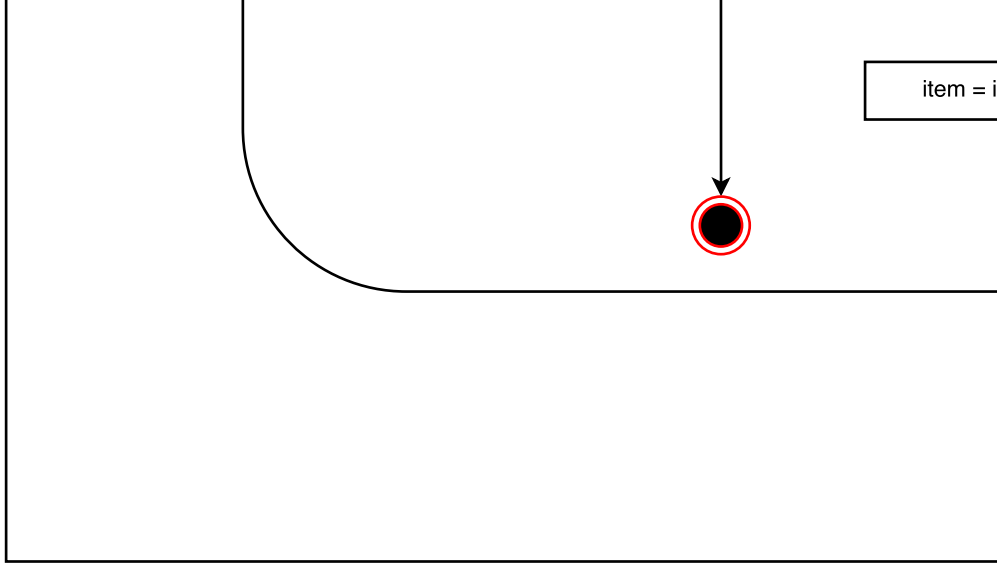

myData[{Resource.Name}] != null
myData["ressources"] != null
ELSE
THROW to parent level
NEW EXCEPTION
Read Received Data
Client == null
FAL
DB == null
FAL
Collection == nu
FAL
THROW to parent level
NEW EXCEPTION

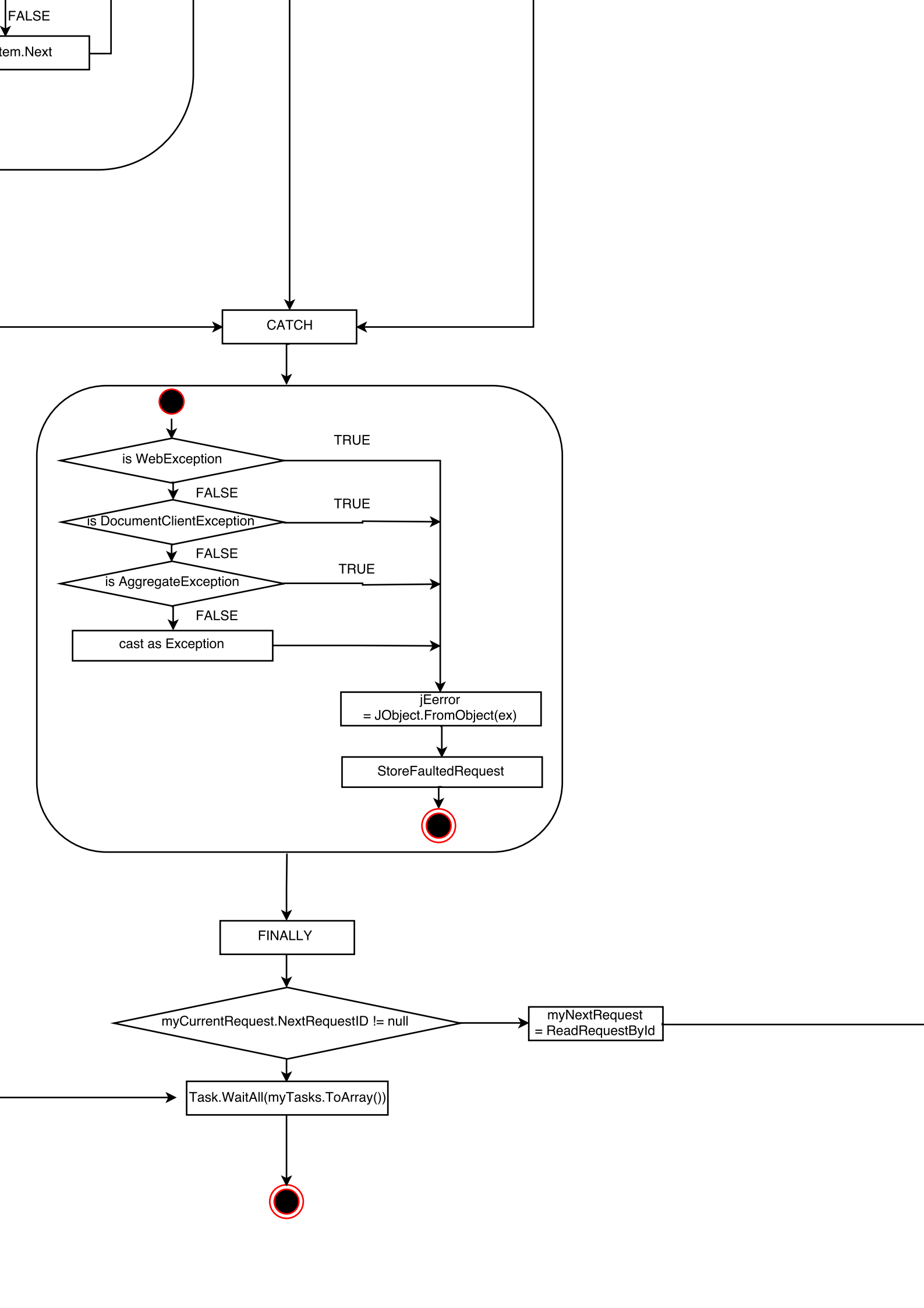
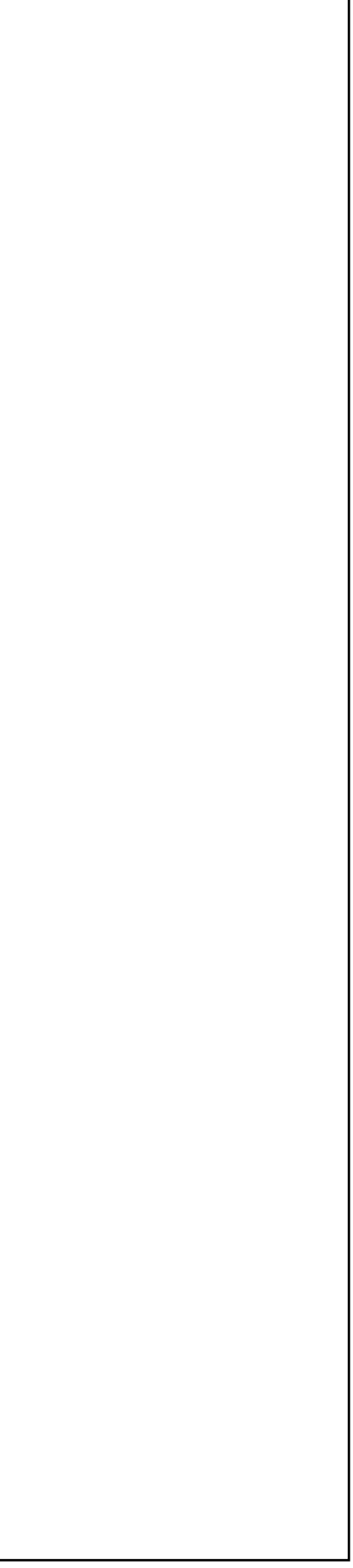

:LaSDScheduler
obtenir la dernière requête active: myNextRequest
myNextRequest != null
TRUE
TRY
myCurrentRequest = myNextRequest
GetDataFunction
myNextRequest.Status <=3
TRY
TRY: sendRequest
Success
TRUE
NormalizeData
FALSE
CATCH
is WebException
FALSE
TRUE
WebExceptionStatus.ProtocolError
FALSE
TRUE
RTEProtocoleErrorAsync
Status == 401
TRUE
FALSE
Status == 408
TRUE
FALSE
Status == 413
TRUE
FALSE
Status == 429
TRUE
FALSE
Status == 503
TRUE
FALSE
Status == 400
TRUE
FALSE
Status == 403
TRUE
Storage.CreateJObject


myNextRequest.Status++
ManageAuthorizationError
ManageTooLargeRequest
ManageTooLargeRequest


.CosmosDB
dispatch
Task.
RessourceRespon
Document
TRY
return
Update myCurrentRequest
dispatch
ReplaceRequest
return
AWAIT
Success
CATCH
is AggregateException
StatusCode == HttpStatus
list = Div
FOR LaSDD
creat
CreateDoc
continu
createTask.
StorageTa
myTask.Ad
myTask.Add
TRUE
item =


createTask=
CreateDocumentAsync
FALSE
TRUE
Status == 500
FALSE
TRUE
Status == 404
FALSE
TRUE
Else
THROW to parent level
continueTask =
createTask.ContinueWith(
StorageTaskCallBack)
myTask.Add(createTask)
myTask.Add(continueTask)


item = i


FALSE
tem.Next
CATCH
is WebException
TRUE
FALSE
is DocumentClientException
TRUE
FALSE
is AggregateException
TRUE
FALSE
cast as Exception
jEerror
= JObject.FromObject(ex)
StoreFaultedRequest
FINALLY
myCurrentRequest.NextRequestID != null
myNextRequest
= ReadRequestById
Task.WaitAll(myTasks.ToArray())