算法伪代码

data\_flow\_graph=read(json\_file)

new abstract\_data\_list

new operation\_list

//combine operation and its output

for (operation in data\_flow\_graph.operations) do

index=find\_operation(operation)

if index<0 then

new new\_operation

new abstract\_data (operation.output)

abstract\_data.input\_operation.add(new\_operation)

new\_operation.output\_data=abstract\_data

operation\_list.add(new\_operation)

abstract\_data\_list.add(abstract\_data)

else

new\_operation=operation\_list.get(index)

abstract\_data=exist\_operation.output\_data

combine(operation.output,abstract\_data)

end if

new\_operation.add\_input(operation.input)

end for

//deal with data which has no operation to generate (which we call init\_data)

new init\_abstract\_data

for (data in data\_flow\_graph.data) do

if data.getGroupId<0 then

init\_abstract\_data.add(data)

end if

end for

abstract\_data\_list.add(init\_abstract\_data)

//replace all old data with new abstract data in new operations’inputs

for (operation in operation\_list) do

new input\_list

for (data in operation.input) do

if !input\_list.contains(data.getGroup) then

input\_list.add(data.getGroup)

data.getGroup.output\_operation.add(operation)

end if

end for

operation.input=input\_list

end for

//get abstract data flow graph

new abstract\_graph(abstract\_data\_list, operation\_list)

//extract services from abstract data flow graph

new service\_list

for (operation in operation\_list) do

new service

service.operation=operation

service.data=operation.output\_data

service\_list.add(service)

end for

return service\_list